Ben-Gurion University of the Negev
The Faculty of Engineering
The Department of Electrical Engineering

Model Base Deep Learning:

ABLE - Adaptive Ultrasound DNN Aided Beamformer

Dor Haim Shmuel

Amit Moses

Under the supervision of Dr. Nir Shlezinger

July 2022

Contents

1 Introduction **3**

1.1 Ultrasound . . . 3

1.1.1 Medical Uses . . . 3

1.1.2 Technique . . . 3

1.1.3 Pre-process . . . 4

2 Beamforming **5**

2.1 Beamforming Principals . . . 5

2.2 Delay and Sum . . . 5

2.3 Adaptive Beamforming . . . 6

2.3.1 MVDR - Minimum Variance Distortion-less Response . . . 6

2.3.2 EBMV . . . 7

2.3.3 Complexity . . . 8

2.3.4 Ultrasound Beamforming Simulation . . . 8

2.4 DNN Augmented Adaptive Beamformer . . . 9

2.4.1 Introduction . . . 9

2.4.2 Network Architecture . . . 9

2.4.3 Activation Functions . . . 10

2.4.4 Loss function . . . 10

2.4.5 Complexity . . . 11

2.5 Results . . . 12

2.5.1 Validation . . . 12

2.5.2 Results . . . 12

2.5.3 Conclusion . . . 13

3 Custom ABLE simulation **15**

3.1 Motivation . . . 15

3.2 Measurement Model . . . 15

3.3 Custom Network Architecture . . . 16

3.4 Experimental Setup . . . 17

3.4.1 MVDR Implementation . . . 17

3.4.2 Data Acquisition . . . 17

3.4.3 Training procedure . . . 18

3.5 Results and insights . . . 18

List of Figures

1.1 Example of an ultrasound B-mode image. The scan shows a 19-week-old fetus. 3

1.2 Illustration of scan sequences, 1.2a Linear scan, 1.2b Plane Wave and 1.2c synthetic aperture. 4

2.1 (a) DAS block diagram. (b) Beamformer output 6

2.2 Eigen-Value Decomposition. 7

2.3 Bemformers output for multiple targets. DAS is compared with Eigen-based MVDR with sub-array size equal to 32. 8

2.4 Eigen-Based MVDR Scheme 9

2.5 Ultrasound image process 9

2.6 2.6a Three spheres behind plane tissue, 2.6b Beamforming image reconstruction 10

2.7 11

2.8 Data-centric & Model-based spectrum 11

2.9 Schematic overview of ABLE. Using the time aligned array signals as input, an artificial agent calculates an optimal set of apodization weights. 12

2.10 Propose artificial agent - schematic overview of the ABLE neural network. 12

2.11 Resolution and contrast metric 13

2.12 Normalized beamprofiles along a simulated pointscatterer for (left) Plane-wave imaging and (right) Synthetic Aperture (IVUS) imaging 13

2.13 Reconstructed Synthetic Aperture in-vivo images using: a) DAS with Hanning apodization, b) iMAP2 c) ABLE, and d) EBMV. 13

2.14 Number of FLOPs for beamforming methods Vs number of receiving input channels. 14

3.1 Uniform Linear Array 15

3.2 Narrow-band signal over the frequency domain 16

3.3 Complex to Real ABLE Architecture 17

3.4 Constellation & Training parameters 18

3.5 Comparison between ABLE, DAS and MVDR for target located at $\theta = 0^\circ$ 18

Chapter 1

Introduction

Biomedical Imaging techniques utilizing various of array processing methods, emphasizing beamforming mechanism in order to enhanced the reconstructed image quality. While advanced beamforming methods can recover much higher image quality than traditional approaches, they are computationally complex and time-consuming. In ultrasound imaging, this complexity extremely significant, striving for efficient, real time and low-cost systems. In this work, we present model-base deep learning method that adopt the algorithmic structure and constraints of adaptive signal processing techniques.

DNN-aided models can efficiently learn to perform high quality ultrasound beamforming while significantly reducing the algorithmic model based complexity. In the work, we will present the classical beamforming methods, name their differences and down-side in simulations, and compere them with the model based deep learning method.

Finally, motivating by the presented model based deep learning beamformer, we will implement an adjustable DNN-aided beamformer over different array processing scenario and examine its performances.

This work influenced from [1] and [2].

1.1 Ultrasound

1.1.1 Medical Uses

For decades, **Ultrasound** imaging has been used for diagnostic purposes. Using ultrasound, the interior structures of the human body can be displayed in real-time and without the use of ionizing radiation. Furthermore, ultrasound imaging benefit from being in-expensive, relatively portable, and it provides very little discomfort for the patient.

Ultrasound examinations commonly known from fetal scans during pregnancy, as seen in Fig 1.1.

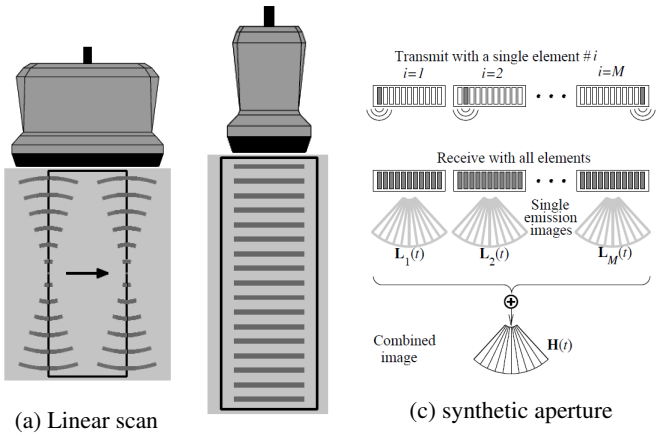
Figure 1.1: Example of an ultrasound B-mode image. The scan shows a 19-week-old fetus.

1.1.2 Technique

Ultrasound scan is carried out by emitting high-frequency acoustic wave fields into the body of the patient. These acoustic waves are in the MHz range and thus, out of the human audible range. Due to changes in the tissue characteristics, the acoustic waves are reflected at different depth within the human body.

The reflected echoes are measured by an array of piezo-electric sensors. By combining these sensor signals using array processing techniques, the distances and intensities of the echoes form the ultrasound B-mode image.

Ultrasound imaging can be carried out using very different imaging sequences: Linear array scan, Plane wave emission and synthetic aperture.



(a) Linear scan

(b) Plane Wave

(c) synthetic aperture

Figure 1.2: Illustration of scan sequences, 1.2a Linear scan, 1.2b Plane Wave and 1.2c synthetic aperture.

- **Linear array scan:** carried out using a linear array transducer, where a small part of the aperture is used for transmitting a narrow beam with a transmitting focus point. Sliding the active part of the aperture across the array, an image consisting of several lines can be created, as illustrate in 1.2a.
- **Plane wave emission:** A plane wave (PW) emission can be achieved by using all elements as both the transmitting and receiving aperture. In this way, the entire image region can be covered in a single emission. Illustrate in 1.2b.
- **Synthetic aperture:** A single transducer element is active, emitting a spherical wave in all directions, covering the entire image region. The active element is then moved along the array, while all elements are used for receiving the back scattered echoes.

1.1.3 Pre-process

The sensing techniques produce RF signal which requires reprocessing before it can be utilized for image reconstruction. (due channel and sampling effects). Analog time-gain compensation is typically applied followed by an anti-aliasing filter (low-pass) and then A/D conversion. Following A/D conversion, high-pass filter is necessary in order to avoid high values in the beamforming summing procedure due low low frequencies.

In addition, interpolation performed up-sample the signal, to achieve more accurate beamforming delays.

Finally, apodization is applied before the beamformer. The entire process of the Ultrasound beamformer illustrate in 2.5, where the prepossess part shown before the beamforming.

Chapter 2

Beamforming

2.1 Beamforming Principals

Array signal processing gathers multiple methods which aims to **gain information** by processing data acquired by collection spatial sensors. those methods are primarily used to:

- Enhance the signal-to-noise ratio
- Characterize a signal in time and space, e.g. by determining the number of emitting sources, their locations or signal parameters.
- Track moving sources.

Beamforming is a technique for directional signal focusing, which performed by applying destructive and constructive interference at a desired direction - Steering. Using this approach we can estimate parameters of an impinging sources.

Beamforming **general concept** is about steering the wave field at different directions and summing the sensor signals, in order to yield maximum output power at the direction of the incoming wave, due to in-phase summation. In 2.1 the beamformer output for an incoming wave incident at 35 degrees is shown. An inherent compromise between the main-lobe width and the side-lobe level exist, respectively to the predefined apodization weights.

2.2 Delay and Sum

Delay-and-Sum maximize power output by delaying, weighting, and summing the sensor signals. It is a method to choose the Synchronizing the received observations at each sensor is done by summing the delayed observations and multiply them by data-Independent weights.

In ultrasound beamforming, the received channel signals are first **time-of-flight** corrected by applying delays according to the geometry of the transducer and imaging scene. Pixel-wise time delays for each array channel can be calculated as:

$$\tau[x,z] = \tau_m(\hat{r}_p) = \frac{||\hat{r}^{Tx} - \hat{r}_p|| + ||\hat{r}_m^{Rx} - \hat{r}_p||}{c} \tag{2.1}$$

Where $\tau_m(\hat{r}_p)$ is the required channel delay to focus to an imaging point r_p, $\hat{r}^{Tx}$ and $\hat{r}_m^{Rx}$ are the positions of the transmitting and receiving element in the array and c is the speed of sound in the medium.

By dynamically focusing to different imaging points, corresponding to pixels in the final image, a pixel-wise response can be generated for every receiving array element $n \in 1, ..., N$, giving by:

$$\mathbf{y}_n[x,z] = \mathbf{x}_n[\tau[x,z]] \tag{2.2}$$

Where $\mathbf{x}_n[t]$ denotes the channel signal for the nth element. A DAS-beamformed image $P_{DAS}[x,z]$ is constructed by multiplying the focused contributions of each receiving channel with predetermined weights $w[x,z]$, and summing the result:

$$\mathbf{B}_{DAS}[x,z] = \mathbf{w}^H[x,z]\mathbf{y}[x,z] \tag{2.3}$$

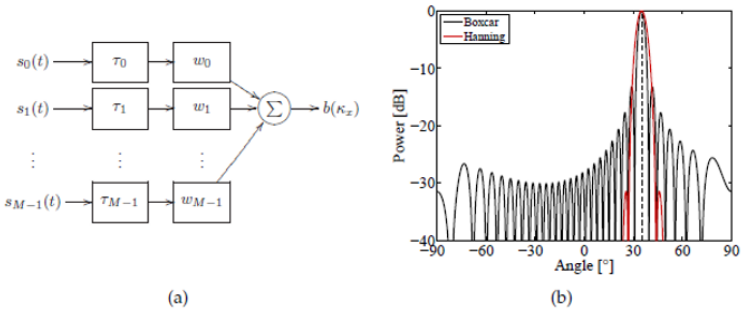


Figure 2.1: (a) DAS block diagram. (b) Beamformer output

A block diagram illustrating the concept of DAS beamforming is shown in 2.1 with the beamformer output for a plane wave incident at 35 degrees using Boxcar and Hanning apodization. The M sensor signals are delayed, weighted and subsequently summed to form the beamformer output.

2.3 Adaptive Beamforming

Integrating beamforming over real time applications, requires to operate under non-ideal conditions such as jammers and noisy environment.

Consequentially, the target signal may be masked by strong Interference which can blind sensors in multiple directions, especially when desired signal is received by a side-lobe. Such scenarios are very challenging for conventional beamformer wheres the weighting procedure is pre-defined and independent in the scenario observations.

In contrast to phase shift beamforming, adaptive beamforming aims to actively assign set of apodization weights, which associated with gained signal's data. The weights which intended for each pixel in the image are directly dependent on the input data.

The usage of adaptive beamformers in medical ultrasound provides improvements of the image quality, in terms of lateral resolution and contrast. One of the widely used method is the **Minimum Variance** beamformer.

2.3.1 MVDR - Minimum Variance Distortion-less Response

The aim of the Minimum Variance Distortion-less Response bemformer is not to directly amplify the power as the desired direction, but to **continuously** updates the weights so that the **power** of the beamformer is **minimized**, while the response from the target point is passed **without distortion**.

The output of the MVDR beamformer is given by:

$$\mathbf{B}_{MVDR}[x,z] = \mathbf{w}^H[x,z]\mathbf{y}[x,z] \tag{2.4}$$

The power of the beamformer output which is also the variance, is given by the following expression:

$$\mathbf{P}_{MVDR}[x,z] = \mathbf{E}\{|\mathbf{B}_{MVDR}[x,z]|^2\} = \mathbf{w}^H[x,z]\mathbf{E}\{\mathbf{y}[x,z]\mathbf{y}^H[x,z]\}\mathbf{w}[x,z] = \mathbf{w}^H[x,z]\mathbf{R}_\mathbf{y}[\mathbf{x},\mathbf{z}]\mathbf{w}[x,z] \tag{2.5}$$

Where **R** denotes the covariance matrix of the pre-steered observations.

Hence, obtaining the MVDR beamformer optimal weights vector is equivalent to solving minimization problem with the following objective function and distortion-less restriction:

$$\begin{aligned} \min_{w} \quad & \mathbf{w}^H[x,z]\mathbf{R}_\mathbf{y}[\mathbf{x},\mathbf{z}]\mathbf{w}[x,z] \\ \text{s.t.} \quad & \mathbf{w}[x,z]\mathbf{a} = 1 \end{aligned} \tag{2.6}$$

where **a** is the steering vector which represents the set of phase delays a plane wave experiences, with respect to reference element in sensors array. Since observations are pre-steered, a is given by vector of ones.

Lagrange multipliers technique yields a closed form solution to the MVDR optimization problem:

$$\mathbf{W}_{MVDR}[x,z] = \frac{\mathbf{R}^-1[x,z]\mathbf{a}}{\mathbf{a}^\mathbf{H}\mathbf{R}^-\mathbf{1}[x,z]\mathbf{a}} \tag{2.7}$$

MVDR solution minimizes the variance and consequentially mitigate the effect of the noise. However, closed form solution requires matrix inversion, which is a **time-consuming** and **unstable** operation. Moreover, the solution is given only if $\mathbf{R}^{-1}[x,z]$ provided. Since the truly statistics of the measurements is **unknown**, the covariance matrix achieved by the **Sample covariance matrix estimator**, which is estimated from the data, and performs poorly under non-ideal conditions.

Numerous methods are proposed in order to avoid the unstable numerical calculation, by increasing the rank of the matrix and supply more realizations od data for the estimation procedure.

Spatial smoothing is a pre-processing algorithm which intended to increase the rank of the covariance matrix R, by **dividing** the N element Uniform linear array into K overlapping sub arrays, each containing $N_{sub} = N - K + 1$ elements. Hench, the required realizations obtained by dividing the data from a single acquisition into a number of subgroups.

The sub-array gained observations $\mathbf{y}_k[x,z]$ is given by:

$$\mathbf{y_k}[\mathbf{x},\mathbf{z}] = [y_k[x,z], y_{k+1}[x,z], ..., y_{k+N_{Sub}-1}[x,z]] \tag{2.8}$$

For each sub-array, a sub-covariance matrix is estimated, where the smooth sample covariance matrix is formulated as the average off the sub-covariance matrices:

$$\mathbf{R}[x,z] = \frac{\sum_{k=0}^{N_{Sub}+1} \mathbf{y}_k[x,z]\mathbf{y}_k^H[x,z]}{N_{Sub}} \tag{2.9}$$

Diagonal Loading method intended to robustify the beamformer by adding small scalar value, ϵ, to the diagonal of the covariance matrix. The value of ϵ is wisely chosen as linear function of the diagonal elements sum multiplied by small value **D** :

$$\epsilon = D * trace(R) \tag{2.10}$$

Where it can be noted that such diagonal addition corresponding to independent noise addition on each of the sensor signals:

$$\mathbf{R_{DL}} = \mathbf{R_y} + \epsilon * \mathbf{I} \tag{2.11}$$

2.3.2 EBMV

Using the assumption that acquired observations can be represented as the sum of the steered received signal and array noise:

$$\mathbf{y}[x,z] = \mathbf{as}[x,z] + \mathbf{n}[x,z] \tag{2.12}$$

Consequentially, it can be defined that the measurements comprised from contributions of elements from both the signal and noise subspace. applying eigen-value decomposition yields the following form of the covariance matrix:

$$R_X = (u_1|...|u_M|u_{M+1}|...|u_N)\begin{pmatrix} \lambda_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda_M & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_{M+1} & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda_N \end{pmatrix}(u_1|...|u_M|u_{M+1}|...|u_N)^H$$

Figure 2.2: Eigen-Value Decomposition.

The M eigenvectors which associated with the M highest eigenvalues defines the **Signal** subspace U_S, (marked in green). The N-M eigenvectors which associated with the N-M equal to noise level eigenvalues, defines the **Noise** subspace (marked in red)

Since the covariance matrix R_X is *Hermitian*, the **Noise** and **Signal** sub-spaces are orthogonal:

$$\mathbf{U_S} \perp \mathbf{U_N} \tag{2.13}$$

Hence there are no **Signal** elements in the **Noise** subspace and vice-versa. Exploiting this fact, we can project the MVDR weights vector onto the **Signal** subspace such that the we eliminate Noise products contribution from the overall beamformer output:

$$\mathbf{W}_{EBMV} = \mathbf{U_S}\mathbf{U_S}^H\mathbf{W}_{MV} \tag{2.14}$$

Eigen-Based MVDR significantly improves the beanmformer performences, at a cost of higher computational complexity, as presented in section XXX.

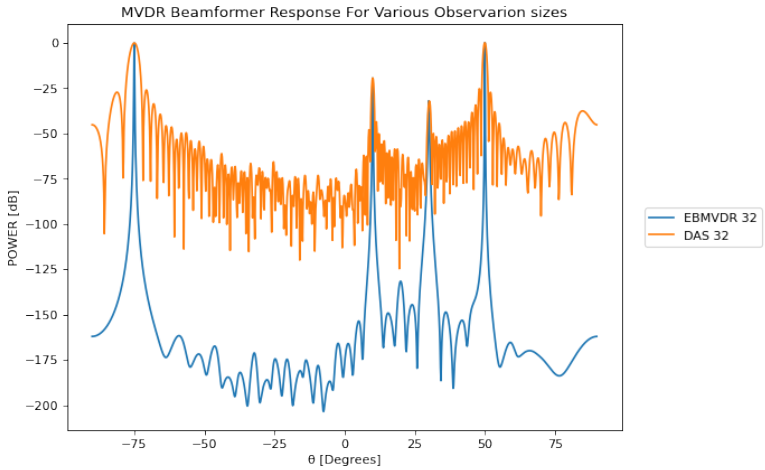


Figure 2.3: Bemformers output for multiple targets. DAS is compared with Eigen-based MVDR with sub-array size equal to 32.

2.3.3 Complexity

Integrating adaptive beanformers into the field of Biomedical Imaging enforce having high-resoulotion but also low computational complexity algorithm. Computational complexity of an algorithm is the amount of time required to run it. Floating-point operations per second, or FLOPS, is the unit of measurement that calculates the performance capability of a supercomputer.

MVDR computational complexity is mostly dictated by the operation of covariance matrix inversion, which requires about N^3 FLOPs, as mentions in [3], In addition to weight calculation operations which sums up to $2N^2 + 3N$ FLOPs. Overall computational complexity of MVDR is approximated by:

$$\mathbf{C_{MV}} \approx N^3 + 2N^2 + 3N \sim \mathbf{O}(N^3) \tag{2.15}$$

Eigen-Based MVDR computational complexity rely on eigen-decomposition and on projection operations complexity which approximately adding N^3 and $(k+1)N^2$ FLOPs, respectively, where k is the ratio of projected eigenvectors. Overall computational complexity of EB-MVDR is approximated by:

$$\mathbf{C_{EBMV}} \approx \mathbf{C_{MV}} + N^3 + (k+1)N^2 \tag{2.16}$$

It can be noted that despite the outstanding performances of the above adaptive beamformers, their long computational time constitutes a burden in real-time applications.

2.3.4 Ultrasound Beamforming Simulation

In order to emphasize the differences between the DAS and EBMV bemaformers, we generated an ultrasound beamforming simulation, the performs the pre-processing procedure at 2.5 and applying beamforming method to reconstruct an image. This proof-of-concept simulation help us to observe two main characteristics - image quality and running time.

We used simulate bulk tissue contrast - three spherical regions with increased scattering and impedance are defined. The centers of the three spheres are co-aligned with center of the transducer in the elevation direction. A schematic of the simulation layout and a slice through the scattering phantom in the beam plane are shown in figure 2.6a.

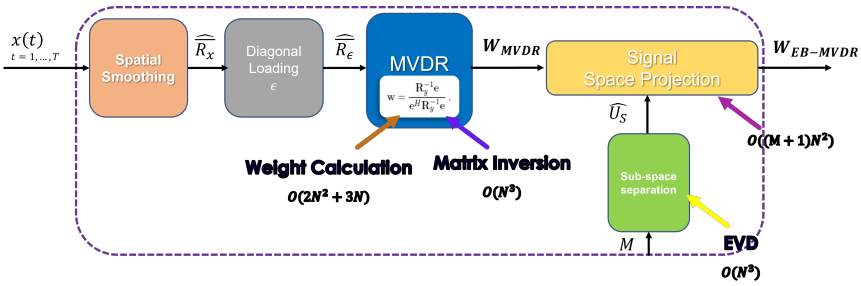


Figure 2.4: Eigen-Based MVDR Scheme

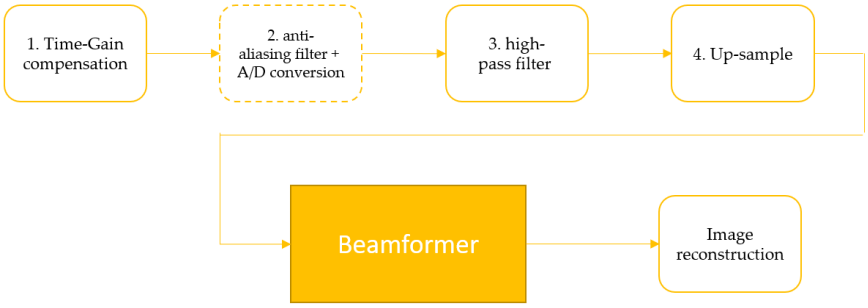


Figure 2.5: Ultrasound image process

The simulation results, which shown in figure 2.6b, present the differences between the different beamforming methods. We can clearly notice that the MVDR produce higher image contrast and less distortions.

MVDR algorithm calculate the beamformer weights for each input time sample. Using the same weights for each time sample is allowedby considering them as i.i.d samples that described the channel statistics for the covariance matrix $R_y = E\{yy^H\}$.

This will lead to much less weights to calculate, leading to less complex model. However, the image reconstruction will suffer from degradation in the form of less contrast and more distortion, as we can see in figure 2.7.

The results show us that the **run-time** extremely decreased as the size of time-sample-group, which lead to degrade in performances.

2.4 DNN Augmented Adaptive Beamformer

2.4.1 Introduction

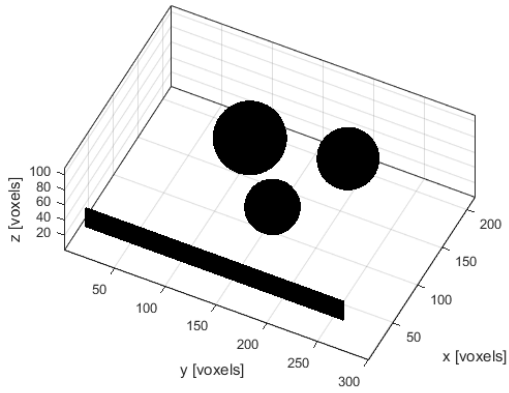
As we discuss in section 2.3.3 the computational complexity of MVDR and EBMV is extremely high, in terms of Ultrasound imaging. In this section we utilize deep learning aided hybrid inference to reduce the bemformer complexity. DNN Aided infertence is a model-base deep learning method 2.8 utilized for robustifying and relaxing the dependence of model-based inference structure tailored for the problem. Model-based algorithm specific computations are replaced with dedicated compact DNNs.

2.4.2 Network Architecture

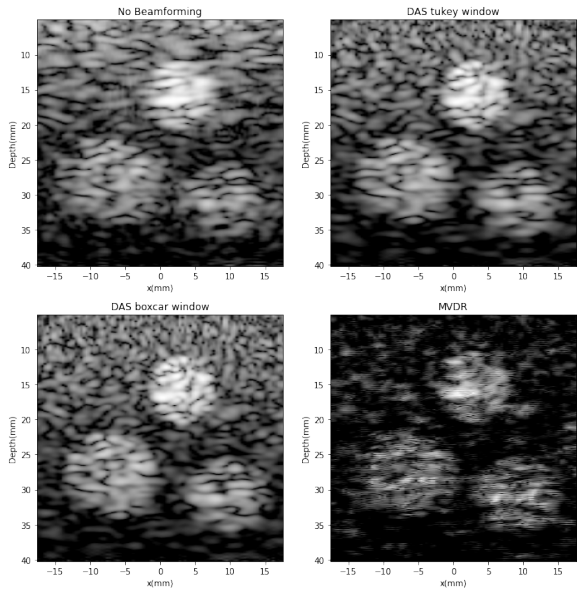
Adaptive Beamforming by deep learning (ABLE) architecture is inspired by the MV beamformer, with neural network that adaptively calculates apodization weights according to the input channel data. Thus, the traditional computationally expensive adaptive processor is replaced with a more computationally efficient artificial agent in the form of neural network, as shown in Figure 2.9. The time-aligned array signals are subsequently multiplied by the predicted weights and summed to yield a beamformed output signal.

A beamformed pixel is obtained by multiplying the time-of-flight corrected array channel data for that pixel with the apodization weights as produced by the neural network:

$$\mathbf{P}_{ABLE}[x,z] = \mathbf{f}_\theta(\mathbf{y}[x,z])\mathbf{y}[x,z] \tag{2.17}$$



(a)



(b)

Figure 2.6: 2.6a Three spheres behind plane tissue, 2.6b Beamforming image reconstruction

ABLE aims not to directly improve the performances of the exist closed form solution, but to produce DNN augmented algorithm which significantly reduce the computational burden and performs just as well as the MVDR

The paper proposed using fully connected (FC) layers which operate on a per-pixel basis, instead of convolutional network which are not effective since spatial invariant is not preserved due changing focus point location.

The proposed network f_θ consists of 4 fully connected (FC) layers comprising N nodes for the outer layers, corresponding to the receiving aperture size, and N/4 nodes for the inner layers, as indicated in figure 2.10 Each layer the number of output nodes is denoted as a function of the aperture size N. Note that the anti-rectifier outputs double the amount of nodes of the preceding fully connected layers.

The **Encoder-decoder** form of the FC layers concatenation **forces** the network to find a **compact representation** of the data, aiding in suppression of noise and incoherent signals.

2.4.3 Activation Functions

The data we working with in RF signals, which contain positive and negative values. RELU activation, ignores negative part and lead to 'dying' nodes. Hence, the paper proposed to use anti-rectifier layers, which combine a sample wise l_2 normalization with the concatenation of the positive and the negative part of the input. This operation is described as:

$$g(\mathbf{x}) = \begin{bmatrix} max\left(0, \frac{\mathbf{x}-\mu_x}{||\mathbf{x}-\mu_x||_2}\right) \\ max\left(0, -\frac{\mathbf{x}-\mu_x}{||\mathbf{x}-\mu_x||_2}\right) \end{bmatrix} \tag{2.18}$$

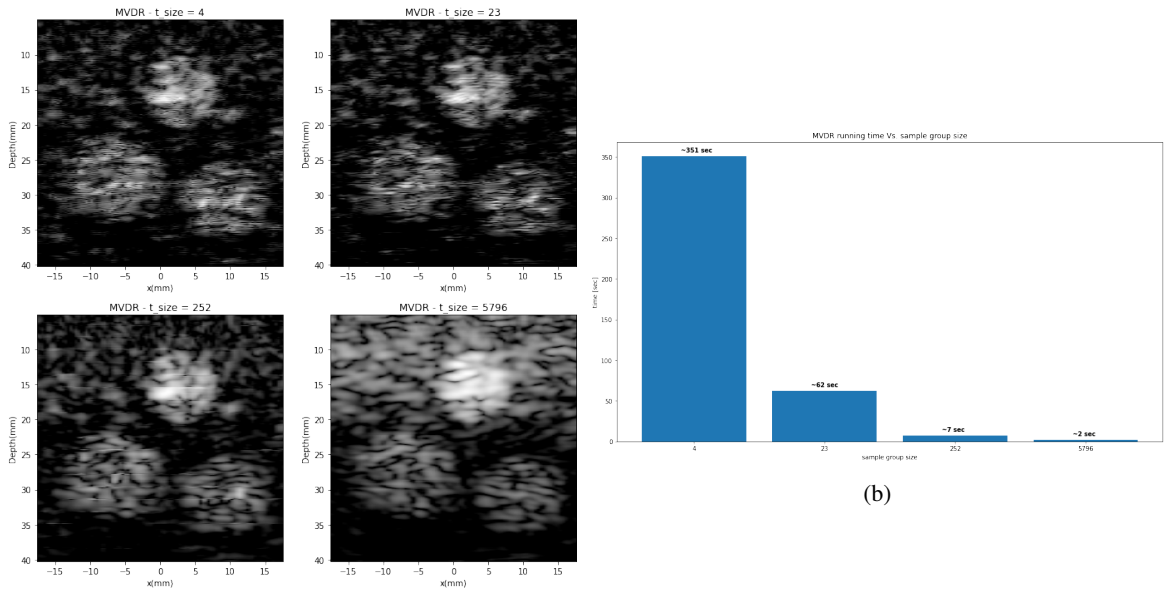
where μ_x denoted the mean of $\mathbf{x}$. The anti-rectifier effectively introduces non-linearity, while preserving the contributions of negative signal components as well as the dynamic range of the input.

2.4.4 Loss function

The total training loss comprised from comparison to MVDR performances and from apodization weight penalty:

- **Image loss:** The image loss comparing between the target image (produced by the MVDR processing) and the image produced using ABLE. The loss using logaritmic compress due to the large dynamic scale of the image. Moreover, the beamformer output is split to negative and positive parts, in order to compere them separately -each by mean-square-logarithmic error, resulting:

$$L_{SMSLE} = \frac{1}{2}||log_{10}(\mathbf{P}^{+}_{ABLE}) - log_{10}(\mathbf{P}^{+}_{MV})||_2^2 + \frac{1}{2}||log_{10}(\mathbf{P}^{-}_{ABLE}) - log_{10}(\mathbf{P}^{-}_{MV})||_2^2 \tag{2.19}$$



(a)

(b)

Figure 2.7

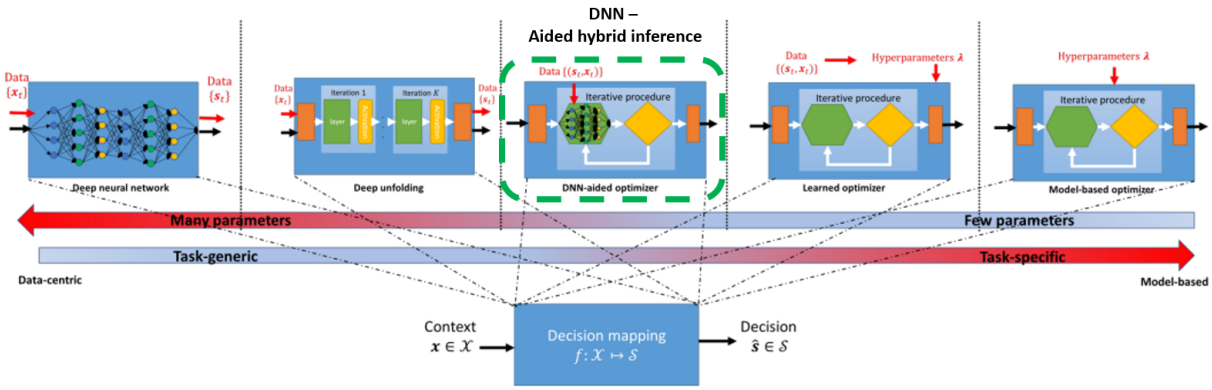


Figure 2.8: Data-centric & Model-based spectrum

- **Weight penalty:** Inspired by MV beamforming, the paper promote a distortionless response by penalizing deviations from unity gain using:

$$L_{unity} = |\mathbf{1}^T f_\theta(\mathbf{y}[x,z]) - 1|^2 \tag{2.20}$$

By considering both the loss function, we get the total loss as:

$$L_{Total} = \lambda L_{SMSLE} + (1 - \lambda L_{unity}) \tag{2.21}$$

where $\lambda \in [0,1]$ is a hyperparameter which determines the weight ratio of the losses.

2.4.5 Complexity

The complexity of the ABLE beamforming calculated during inference only. Every layer could be described as matrix product with added bias. For matrix with dimensions $N_i x N_{i+1}$ and bias with N_{i+1} elements, we gets $2N_iN_{i+1} + N_{i+1}$ FLOPs. The activation require double the FLOPs RELU require, spo 4 FLOPs in total. The total amount of FLOPs is:

$$C_{ABLE} = 2N_0N_1 + N_1 + \sum_{i=1}^{L-1} 4N_iN_{i+1} + N_{i+1} + 4Ni \tag{2.22}$$

We can notice that the ABLE complexity is about $O(N^2)$

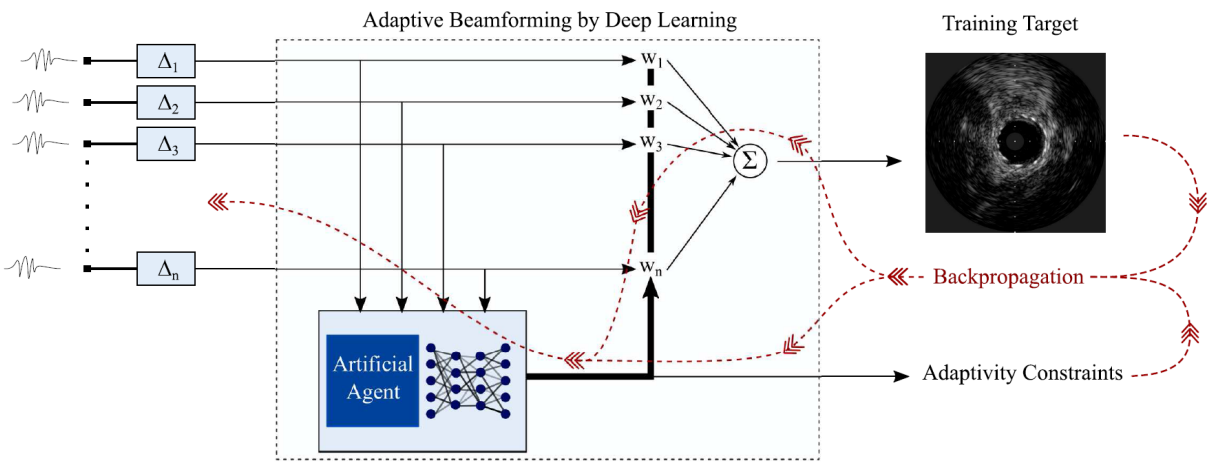


Figure 2.9: Schematic overview of ABLE. Using the time aligned array signals as input, an artificial agent calculates an optimal set of apodization weights.

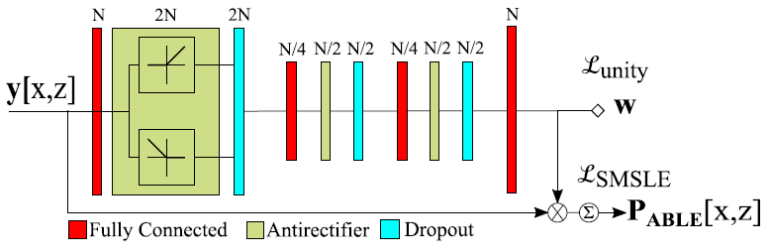


Figure 2.10: Propose artificial agent - schematic overview of the ABLE neural network.

2.5 Results

2.5.1 Validation

The paper introduce the follwing evaluation metrics:

- FWHM: Resolution was assessed by evaluating the Full-Width-at- Half-Maxima (FWHM) of simulated point scatterers, which were calculated for both the lateral and the axial direction.
- CNR: Contrast of the images was measured by the contrastto- noise ratio (CNR) between a low and high intensity region.

$$CNR = 10log_{10}\left(\frac{|\mu_{low} - \mu_{high}|}{\sqrt{(\sigma^2_{low} + \sigma^2_{high})/2}}\right) \tag{2.23}$$

  where μ_{low} and μ_{high} represent the mean intensity of the low and high intensity regions, respectively, and $sigma^2_{low}$ low and $sigma^2_{high}$ high the variance of the low and high intensity regions, respectively.

- MAE: mean-absolute-error (MAE), between the target images (EBMV) and the other imaging modalities

2.5.2 Results

We can notice in 2.11 that ABLE outperforms conventional DAS and iMAP2 beamforming. Their beam profiles are visualized in 2.12. Interestingly, ABLE even yields better lateral resolution than its train target, EBMV, displaying the power of mean-error optimization across many examples.

From the simulated point scatters in 2.13 one can see that DAS beamforming yields wider and more pronounced sidelobes than iMAP2, ABLE and EBMV. Also, strong clutter suppression for ABLE and EBMV compared to DAS. Images in 2.13

Plane wave (linear array)	DAS	iMAP2	ABLE	EBMV
$FWHM_{lat}$ (mm)	0.846 ± 0.10	0.730 ± 0.12	0.704 ± 0.16	0.778 ± 0.14
$FWHM_{ax}$ (mm)	0.431 ± 0.006	0.430 ± 0.008	0.218 ± 0.069	0.434 ± 0.013
CNR (dB)	11.35 ± 1.45	9.59 ± 2.81	11.91 ± 1.03	12.67 ± 1.04
MAE	9.86 ± 4.15	3.37 ± 1.41	3.29 ± 1.48	-

Synthetic Aperture (circular array)	DAS	iMAP2	ABLE	EBMV
$FWHM_{lat}$ (rad)	0.127 ± 0.003	0.094 ± 0.004	0.055 ± 0.009	0.087 ± 0.008
$FWHM_{ax}$ (mm)	0.217 ± 0.021	0.173 ± 0.039	0.118 ± 0.026	0.046 ± 0.025
CNR (dB)	11.21 ± 3.42	10.09 ± 3.00	13.98 ± 1.10	13.62 ± 0.8109
MAE	19.58 ± 0.53	6.05 ± 1.77	0.65 ± 0.18	-

Figure 2.11: Resolution and contrast metric

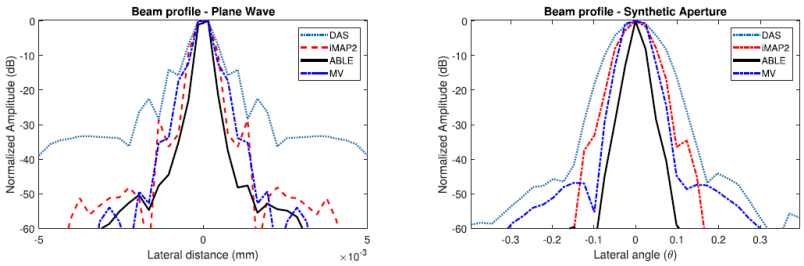


Figure 2.12: Normalized beamprofiles along a simulated pointscatterer for (left) Plane-wave imaging and (right) Synthetic Aperture (IVUS) imaging

are logarithmic compressed with a dynamic range of 60dB. From top to bottom the we have a wire phantom, an arterial stent, and a coronary artery with plaque.

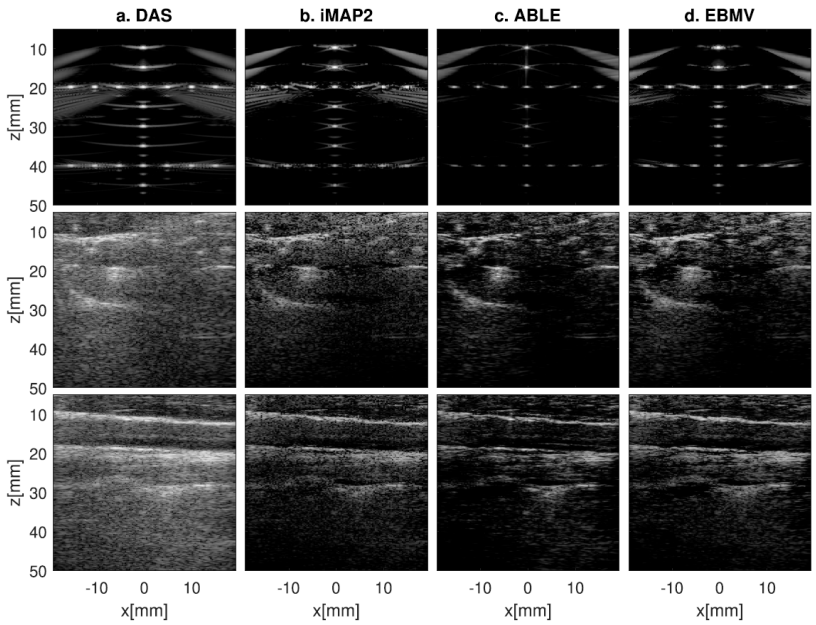


Figure 2.13: Reconstructed Synthetic Aperture in-vivo images using: a) DAS with Hanning apodization, b) iMAP2 c) ABLE, and d) EBMV.

2.5.3 Conclusion

One of the noticeable property of the ABLE, is its ability to generalize various types of data, even is it is not included in the training phase. Morover, ABLE learn from the MV beamformer how to suppression of typical ringdown artifacts caused by residual vibrations after transmission.

Figure 2.14 show the number of FLOPs required for each beamformer as a function of the input array size The model-base deep learning approach provides a lower computational cost than the MV based methods relying on matrix inversion.

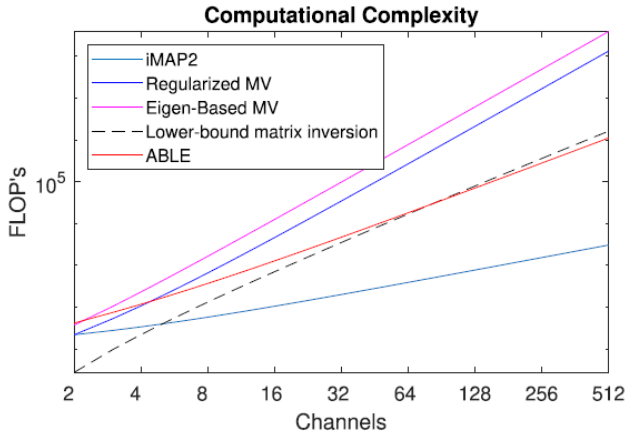


Figure 2.14: Number of FLOPs for beamforming methods Vs number of receiving input channels.

This approach provide reduces computational complexity, in particular for a large number of transducer elements. Large array will benefit more from the ABLE. For example, 3D ultrasound systems that obtain 1000 elements.

Chapter 3

Custom ABLE simulation

3.1 Motivation

The proposed deep learning-based adaptive processing framework can contribute in many signal processing applications. Moreover, the usage of DNN-aided inference is being asked in applications that own a model-based closed form solution, such that performance degrades only over extreme scenarios. In that case, data efficiency is recommended.

Inspired by the authors of [2] vision, and since Biomedical data sets such as MRI or Ultrasound scans are not available in large amounts such that required for a data-driven model (only few were found), we decided to adopt the ABLE adaptive beamformer into custom system model, where data could synthetically created.

The problem we wish to deal with using an implementation of ABLE, is the task of direction of arrival beamforming, which is a crucial task in array processing with a wide range of applications, such as radar, communication and etc.

Using an array of sensors intended for signals acquisition, we aim to accurately locate the angel of arrival of targets, and shape the receiving array response in order to amplify the beam output at the desired direction (equivalently for jammers and transmitting output).

3.2 Measurement Model

Consider Uniform Liner Array of sensors comprised from **N** antennas with spacing of d between elements, acquiring measurements $\mathbf{X(t)} = [X_1(t), ..., X_N(t)]$ from **M** impinging signals located at $\theta = [\theta_1, ..., \theta_M]$.

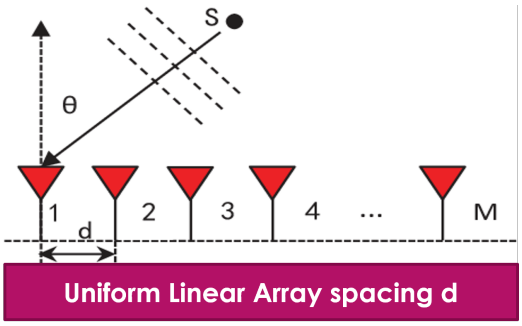


Figure 3.1: Uniform Linear Array

The measurements **X(t)** acquired as a linear combination of incident **signals** $\mathbf{S(t)} = [S_1(t), ..., S_M(t)]$ and the antennas **noise** $\mathbf{V(t)} = [v_1(t), ..., v_N(t)]$.

The noise $\mathbf{V(t)}$ and the signals $\mathbf{S(t)}$ are i.i.d and uncorrelated Gaussian complex Normal processes.

$$\mathbf{S(t)} \sim \mathcal{CN}(\mathbf{0}, \sigma_s^2 \mathbf{I_M}) \tag{3.1}$$

$$\mathbf{V(t)} \sim \mathcal{CN}(\mathbf{0}, \sigma_w^2 \mathbf{I_N}) \tag{3.2}$$

The signals $\mathbf{S(t)}$ are generated at in the far-field region and the propagation is uniform in all directions. Hence, source

Information lies in the difference between the source propagation delay at the reference sensor $\boldsymbol{\tau}$ and at the $\mathbf{n_{th}}$ sensor, $\Delta\tau_n$.

The measurements acquired at the $\mathbf{n_{th}}$ sensor is given by:

$$\mathbf{x_n(t)} = \alpha_n \mathbf{S}(t - \Delta\tau_n) + \mathbf{v_n(t)} \tag{3.3}$$

And over the Frequency domain it can be represent as:

$$\mathbf{x_n(w_k)} = \alpha_n \mathbf{S}(w_k) e^{-jw_k \Delta\tau_n} + \mathbf{v_n}(w_k) \tag{3.4}$$

Moreover, assuming **narrow-band** signal, the **information** is centred over dominant frequency, such that the measurement model can be represented in the base-band area.

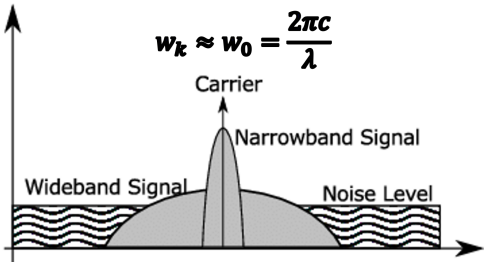


Figure 3.2: Narrow-band signal over the frequency domain

Hence, the received signal at the $\mathbf{n_{th}}$ sensor over the time domain can be expressed as:

$$\mathbf{x_n(t)} = \alpha_n \sum_{m=1}^{M} \mathbf{S}_m(t) e^{-jw_0 \Delta\tau_n} + \mathbf{v_n}(t) \tag{3.5}$$

By setting the spacing between elements over the ULA $\mathbf{d}$ to be half of wavelength $\mathbf{d} = \frac{\lambda}{2}$, for the $\mathbf{m_{th}}$ impinging source, the propagation delay at the $\mathbf{n_{th}}$ sensor is achieved by:

$$\tau_n = \frac{1}{c}(n-1)dsin(\theta_m) = \frac{\lambda}{2c}(n-1)dsin(\theta_m) \tag{3.6}$$

where c is the speed of light, and λ is the wave length. The contribution of all sensors can be expressed as:

$$\mathbf{X}(t) = \sum_{m=1}^{M} \mathbf{a}(\theta_m)\mathbf{S}_m(t) + \mathbf{V}(t) \tag{3.7}$$

where the steering vector $\mathbf{a}(\theta_m)$ is the array response for the $\mathbf{m_{th}}$ impinging source:

$$\mathbf{a}(\theta_m) = [1, e^{-j\pi sin(\theta_m)}, ..., e^{-j\pi(M-1)sin(\theta_m)}] \in \mathbb{C}^{1xM} \tag{3.8}$$

Consequentially, The signal measurement model for **ULA** structure containing **N** antennas measuring **M** impinging M narrow-band sources takes the following form:

$$\mathbf{X}(t) = \mathbf{A}(\theta)\mathbf{S}(t) + \mathbf{V}(t)$$

where the matrix $\mathbf{A}(\theta) \in \mathbb{C}^{NxM}$ comprised from the steering vectors $\{\mathbf{a}(\theta_1), ..., \mathbf{a}(\theta_M)\}$.

3.3 Custom Network Architecture

Simulating custom **ABLE** model for the presented measurements model, requires several changes in the architecture of the network:

1. Our observation input vector $\mathbf{X}(t) \in \mathbb{C}^{Nx1}$ is over the **Complex** domain, while current architecture supports only **Real** input data. Hence, we proposed and tested several permutations of the ABLE architecture, where the most beneficial and time-consuming one, designed such that input and output fully-connected layers **expands**, i.e. both the input dimension of the first dense layer and the output dimension of the last dense layer increased from **N to 2N**, where N is the number array sensors.
2. Tuning parameters of the network such as **Dropout** level and **Weight** penalty, choose differently.

Consequentially, the proposed ABLE architecture is defined in figure 3.3.

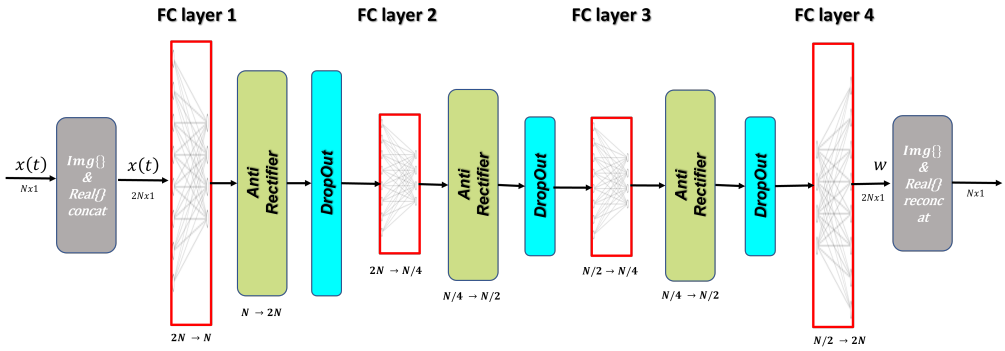


Figure 3.3: Complex to Real ABLE Architecture

3.4 Experimental Setup

3.4.1 MVDR Implementation

Based on 3.3 system model constellation, and in order to create ground truth for both data generation for custom ABLE model training and for models performance comparison, MVDR beamformer is applied.

In analogy to Ultrasound beamforming task, for each **azimuth** angle of arrival θ over the range $\theta \in [-\frac{\pi}{2}, -\frac{\pi}{2}]$, the MVDR weights vector is calculated in the following manner:

$$\mathbf{W}_{MVDR}(\theta) = \frac{\mathbf{R}_X^{-1}\mathbf{a}(\theta)}{\mathbf{a}(\theta)^H\mathbf{R}_\mathbf{X}^{-1}\mathbf{a}(\theta)} \tag{3.9}$$

And the power of the beamformer output is given by:

$$\mathbf{P}_{MVDR}(\theta) = \mathbf{w}_{MVDR}^H(\theta)\mathbf{R}_\mathbf{X}\mathbf{w}_{MVDR}(\theta) \tag{3.10}$$

The power of beamformer is used (instead of beamformer output)since the discussed scenario deals with non-deterministic signals.

3.4.2 Data Acquisition

As described in the ABLE section, The proposed model is trained by comparing its output to the ground truth MVDR one. The generated data set which used for both training and evaluating the custom ABLE architecture, is created similarly to the dataSet presented in the [2]:

First, for each generated signals with pre-determind directions of arrivel θ_M, observation vector $\mathbf{X}(t) \in \mathbb{C}^{Nx1}$ is produced. Afterwards, for each angle in the desired range $\theta_j \in [-\frac{\pi}{2}, -\frac{\pi}{2}]$, the observation vector is steered to yield the following input vector:

$$\mathbf{y}_t(\theta_j) = \mathbf{a}(\theta_j)^H\mathbf{X}(t) \tag{3.11}$$

The steered input vector is injected to the MVDR to generate the bemformer output at the relevant direction $\mathbf{P}_{MVDR}(\theta_j)$. Finally each steered observation vector $\mathbf{y}_t(\theta_j)$ and is coupled with the corresponding MVDR output to crease single sample in the desired data set:

$$\mathbf{D}_{ABLE} = \{(\mathbf{y}_t(\theta_j), \mathbf{P}_{MVDR}(\theta_j))\}_{d=1}^{d=|D|-1} \tag{3.12}$$

3.4.3 Training procedure

For the simulation, we generated the following constellation, as presented in Table XXX:

Parameter	Value
System model Parameters	
Array Size (N)	64
Number of sources (M)	1
SNR	10 [dB]
Range of angels	$R_{angles} = \left[-\frac{\pi}{2}, \frac{\pi}{2}\right], \lvert R \rvert = 360$
MVDR Parameters	
Sub array size	16
Diagonal Loading value	$\epsilon = 0.001$
ABLE Tuning Parameters	
Dropout value	0.5
Weight ratio	$wlp = 0.5, 0.99$
Data Set size	360 * 2500 samples
Diagonal Loading value	$\epsilon = 0.001$
Optimizer	Adam
Learning Rate	10^{-3}
Batch size	360
Weight decay	L2-norm, $\gamma = 10^{-6}$

Figure 3.4: Constellation & Training parameters

We used with the same loss measure as presented in [2].

3.5 Results and insights

We can notice that the Custom ABLE model achieves quite good performances by introducing sharp and low power side lobs, surrounding high main lob in the target direction. we can notice that as expected, ABLE performs better than DAS beamformer and as quite good as MVDR, as presented in figure 3.5.

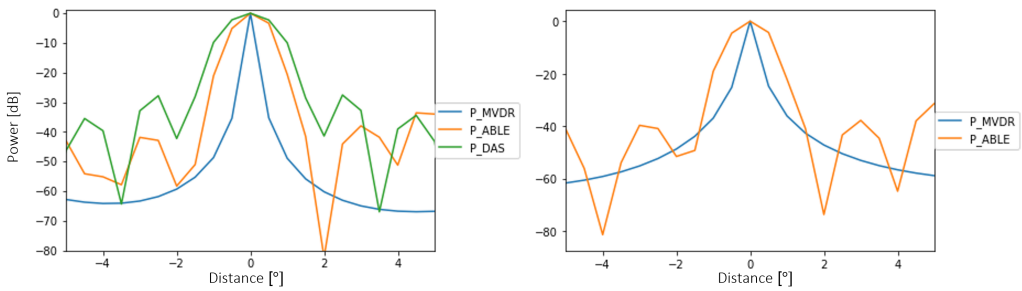


Figure 3.5: Comparison between ABLE, DAS and MVDR for target located at $\theta = 0°$

ABLE model is mainly aimed to reduce the computational complexity burden induced by the MVDR. as presented, the overall computational complexity of MVDR is approximated by: $\mathbf{C_{MV}} \sim \mathcal{O}(N^3)$ while the complexity of custom ABLE is approximated by $\mathbf{C_{ABLE}} \sim \mathcal{O}(N^2)$. Simulations indicates that ABLE model ouput is given after 0.1262 [sec], while MVDR beamformer output is given after 7.09 [sec] , i.e ABLE model is faster than MVDR by **56.2** $\sim$ **N (64)**, as expected.

We can notice that ABLE motivation is applicable not only over Bio-medical imaging field, but it can **fit** to various of challenges in the area of **Array processing**.

Bibliography

[1] I. K. Holfort, *Adaptive beamforming for medical ultrasound imaging*. Technical University of Denmark, DTU Electrical Engineering, 2009.

[2] B. Luijten, R. Cohen, F. J. de Bruijn, H. A. Schmeitz, M. Mischi, Y. C. Eldar, and R. J. van Sloun, "Adaptive ultrasound beamforming using deep learning," *IEEE Transactions on Medical Imaging*, vol. 39, no. 12, pp. 3967–3978, 2020.

[3] R. Lorenz and S. Boyd, "Robust minimum variance beamforming," *IEEE Transactions on Signal Processing*, vol. 53, no. 5, pp. 1684–1696, 2005.