Deep Learning-Based Optimal Spatial Subsampling in Ultrasound Nondestructive Testing

Han Wang[1], Eduardo Pérez[1,2], Florian Römer[1]
[1]*Fraunhofer Institute for Nondestructive Testing*, Saarbrücken, Germany
[2]*Dept. Electronic Measurements and Signal Processing, Technische Universität Ilmenau*, Ilmenau, Germany
han.wang@izfp.fraunhofer.de, eduardo.perez@tu-ilmenau.de, florian.roemer@izfp.fraunhofer.de

***Abstract*—Traditional ultrasound synthetic aperture imaging relies on closely spaced measurement positions, where the pitch size is smaller than half the ultrasound wavelength. While this approach achieves high-quality images, it necessitates the storage of large data sets and an extended measurement time. To address these issues, there is a burgeoning interest in exploring effective subsampling techniques. Recently, Deep Probabilistic Subsampling (DPS) has emerged as a feasible approach for designing selection matrices for multi-channel systems. In this paper, we address spatial subsampling in single-channel ultrasound imaging for Nondestructive Testing (NDT) applications. To accomplish a model-based data-driven spatial subsampling approach within the DPS framework that allows for the optimal selection of sensing positions on a discretized grid, it is crucial to build an adequate signal model and design an adapted network architecture with a reasonable cost function. The reconstructed image quality is then evaluated through simulations, showing that the presented subsampling pattern approaches the performance of fully sampling and substantially outperforms uniformly spatial subsampling in terms of signal recovery quality.**

***Index Terms*—Signal Spatial Subsampling, Signal Reconstruction, Deep Learning, Ultrasound NDT**

I. Introduction

Ultrasound imaging is widely used in Nondestructive Testing (NDT) and medical applications due to its mobility, reliability, availability and cost-effectiveness. However, during the scanning process, satisfying the spatial Nyquist condition often requires a small spacing between elements, which leads traditional sensors to use a large number of measurement positions for covering a given aperture. Such dense sampling can significantly increase the measurement time and impose substantial demands on the hardware for data acquisition, storage, and processing. Therefore, developing advanced spatial subsampling algorithms is of great importance, as subsampled data can generate high-quality images when properly processed [1].

Spatial subsampling refers to the reduction of sensor locations or array channels in data acquisition. This task is closely tied to optimal sensor placement and sparse array design when the spatial domain is discretized into a grid. A spatial subsampling algorithm aims to achieve various objectives, taking into account multiple performance parameters that vary depending on the particular scenario. These parameters typically include either the beam pattern or the image quality. For example, Cohen introduced the sparse nonlinear Convolutional Beamforming Algorithm (COBA), which achieves an improved beam pattern over that of Delay and Sum (DAS) beamforming by using fewer array elements [2]. Diamantaras introduced Learn-to-Select (L2S) using Softmax selection for the choice of antennas from a dense Uniform Linear Array in order to obtain the desired beam pattern [3]. Mamistvalov developed a deep-learning-based approach for reconstructing B-mode images from temporally and spatially subsampled channel data, incorporating the Structural Similarity Index Metric (SSIM) [4] into the loss function [5]. However, this approach only excels in medical images due to challenges such as reverberation, non-linear propagation and aberrating layers in the human body. Moreover, Huijben proposed Deep Probabilistic Subsampling (DPS) algorithm by leveraging the Gumbel-Softmax reparameterization trick to select array channels while ensuring superior reconstructed image quality in medical applications [6]. Similarly, in ultrasound NDT, Compressed Sensing (CS) theory allows for subsampling of the measured signal in both the spatial and frequency domains without sacrificing high-resolution reconstructions [7]. Furthermore, we are doing a localization task with a sensor array where the sensor placement affects the estimation performance, so parallels can be drawn between ultrasound and radar localization [8].

In this study, we focus on practical applications of ultrasound NDT and implement a model-based data-driven spatial subsampling algorithm under the DPS framework. Our technique enables the selection of optimal sensing locations for single-channel synthetic aperture ultrasound imaging. To achieve this, we construct a specific signal model, select an appropriate optimization target and design a tailored neural network with reduced complexity. We accelerate the learning process by exploiting the data model to enforce the required subsampling matrix structure without the need of constraints or penalty terms in the cost function. Additionally, we evaluate its performance by comparing the image quality with that of fully sampled and uniformly spatially subsampled signal recovery.

II. Signal Model and Problem Formulation

For simplicity, we will consider the pulse-echo measurements at a multitude of sensor locations on a spatial grid,

This work was partially supported by the Fraunhofer Internal Programs under Grant No. Attract 025-601128, the German Research Council (DFG) under the project "CoSMaDu", as well as the Thuringian Ministry of Economic Affairs, Science and Digital Society (TMWWDG).

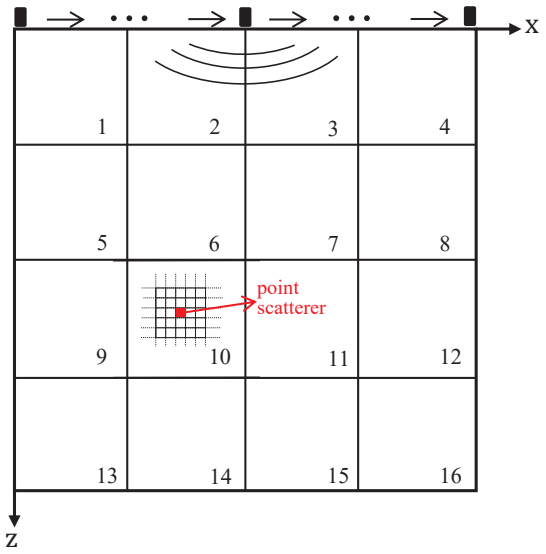

Fig. 1. 2-dimensional measurement design illustration. A transducer starts moving from the origin across the surface of the specimen, transmitting pulses and receiving echoes at pre-determined positions that are equidistant from each other. The 2D specimen is divided into 16 regions for further use in III-C. The whole specimen is discretized into $N_\mathrm{x} \times N_\mathrm{z}$ fine square cells, for example in the region-10, with each cell indicating a possible scatterer location and carrying a scattering amplitude.

which can also be interpreted as a special case of phased array measurements. As shown in Fig. 1, the sensor moves along a specified axis, transmitting and receiving pulses at predefined locations. To represent a received signal $p(t)$, called an A-scan, we model the pulse echo by a real-valued Gabor function. We then assume that each A-scan is a linear combination of time-shifted, energy-attenuated versions of the transmitted pulse function with inverted phase, where each shift is caused by an isolated flaw, as follows [9]:

$$p(t) = \sum_{k=1}^{K} a_k e^{-\alpha(t-\tau_k)^2} \cos(2\pi f_c(t-\tau_k)) + n(t) \tag{1}$$

Here, the lower case k indicates the scatterer index and the upper case K is the total number of scatterers; a_k is the scattering amplitude, which contains information about the reflectivity of the k-th scatterer and the energy attenuation in the medium. The time delay τ_k refers to the pulse propagation time between a transducer and the k-th scatterer. As for the transducer calibration information, we model the center frequency as $f_c = 4$ MHz, and the bandwidth factor as $\alpha = f_c^2$. Finally, the additive white Gaussian noise (AWGN) is added to each received echo and is denoted as $n(t)$.

We assume that the transducer samples at a rate of $f_s = 60$ MHz for an observation period T. So a measurement signal $p(t)$ at one location consists of $N_\mathrm{t} = f_s \cdot T$ samples, which can be rewritten as a vector $\mathbf{p} \in \mathbb{R}^{N_\mathrm{t}}$. After collecting all the A-scans at $N_\mathrm{x} \in \mathbb{N}$ locations along the scan axis, we stack all $N_\mathrm{x} \cdot N_\mathrm{t}$ samples into an observation vector $\mathbf{y} \in \mathbb{R}^{N_\mathrm{x}N_\mathrm{t}}$, which is in form of $[\mathbf{p}_1^\mathrm{T} \quad \mathbf{p}_2^\mathrm{T} \quad \dots \quad \mathbf{p}_{N_\mathrm{x}}^\mathrm{T}]^\mathrm{T}$. Assuming an idealized point scatterer model, we can express $\mathbf{y}$ as:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n} \tag{2}$$

where $\mathbf{x} \in \mathbb{R}^{N_\mathrm{x}N_\mathrm{z}}$ is a vectorized specimen map of $N_\mathrm{x} \times N_\mathrm{z}$ potential scatterer locations. We define $N_\mathrm{x} = 48$ and horizontal grid distance $\Delta\mathrm{x} = 0.5$ mm, and $N_\mathrm{z} = N_\mathrm{t} = 256$, so the vertical grid distance can be computed by $\Delta\mathrm{z} = \mathrm{c}/(2f_s)$, where $\mathrm{c} = 5850$ m/s is the ultrasonic velocity in steel. $\mathbf{A} \in \mathbb{R}^{(N_\mathrm{x}N_\mathrm{t})\times(N_\mathrm{x}N_\mathrm{z})}$ is called the measurement matrix or forward model, whose column $\mathbf{A}_{\cdot,j}$ contains the vectorized and discretized volumetric observation of a single scatterer at vectorized postion j.

In many NDT applications, prior knowledge of the specimen geometry is available. Sometimes, this knowledge can be used to infer regions of interest (ROI) that are of particular relevance to the inspection task, e.g., because defects in that region are critical to the quality assessment of the specimen. In such cases, the spatial sampling can be optimized to prioritize image quality in these regions, allowing critical defects to be located and characterized.

In this case, we design a binary subsampling matrix $\mathbf{S} \in \mathbb{R}^{M\times N_\mathrm{x}}$ to select M locations from N_x, where $M < N_\mathrm{x}$, which can be physically implemented e.g. by a positioner system. The selection matrix $\mathbf{S}$ consists of M one-hot rows $[\mathbf{s}_1 \quad \mathbf{s}_2 \quad \dots \quad \mathbf{s}_M]^\mathrm{T}$, and each row contains N_x elements, for example the m-th row $\mathbf{s}_m = [s_{m,1} \quad s_{m,2} \quad \dots \quad s_{m,N_\mathrm{x}}]$. Then the subsampled data $\mathbf{y}_s \in \mathbb{R}^{MN_\mathrm{t}}$ can be calculated:

$$\mathbf{y}_s = (\mathbf{S} \otimes \mathbf{I})\mathbf{y} = (\mathbf{S} \otimes \mathbf{I})(\mathbf{A}\mathbf{x} + \mathbf{n}) \tag{3}$$

where $\otimes$ denotes the Kronecker product, and $\mathbf{I} \in \mathbb{R}^{N_\mathrm{t}\times N_\mathrm{t}}$ is an identity matrix.

After acquisition, the Synthetic Aperture Focusing Technique (SAFT) [10] can be used to produce a B-mode image. Alternatively, a Sparse Signal Recovery (SSR) algorithm such as the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [11] can be used to obtain a high-resolution reconstructed image. In this paper, we take advantage of the Learned ISTA (LISTA) [12] [13] to execute the SSR of $\mathbf{y}_s$ because LISTA is superior to conventional FISTA in several aspects. Firstly, LISTA has a faster computation speed; secondly, LISTA could have enhanced reconstruction capabilities.

III. Optimal Spatial Subsampling

A. Gumbel-Softmax Reparameterization

To construct the matrix $\mathbf{S}$, we determine the support of each row $\mathbf{s}_m$, which can be thought of as a sample from a categorical distribution. We assume that $\mathbf{s}_m$ follows a categorical distribution, $\mathbf{s}_m \sim \mathrm{Cat}(N_\mathrm{x}, \boldsymbol{\theta}_m)$, where each element $s_{m,i}$ corresponds to an unnormalised probability $\theta_{m,i}$ ($\sum_{i=1}^{N_\mathrm{x}} \theta_{m,i} \neq 1$). Essentially, $\theta_{m,i}$ represents the probability that sensor location i will be the mth selected location.

We consider using neural networks to sample categorical variables; however, networks with discrete variables are difficult to train because the gradient backpropagation algorithm cannot be applied to non-differentiable layers. Therefore,

we leverage the Gumbel-Softmax reparameterization trick to replace non-differentiable categorical samples with a differentiable approximation during training [14] [15]. The unnormalized log-probabilities (logits) are conventionally computed first, allowing the unconstrained optimization of $\log(\boldsymbol{\theta}_m)$. The critical step is to add a standard Gumbel noise vector to the logits, with $\boldsymbol{g}_m \sim \mathrm{Gumbel}(0,1)$, and yielding noisy logits $\mathbf{u}_m = \log(\boldsymbol{\theta}_m) + \boldsymbol{g}_m$. Then the gradient estimation will depend on the realization of the Gumbel noise [16].

Before drawing hard samples from $\mathbf{u}_m$, we apply a $\mathrm{softmax}$ operation to obtain soft samples $\mathbf{z}_m$, whose i-th element is computed by:

$$z_{m,i} = \frac{\exp((\log(\theta_{m,i}) + g_{m,i})/\tau)}{\sum_{j=1}^{N_\mathrm{x}} \exp((\log(\theta_{m,j}) + g_{m,j})/\tau)} \tag{4}$$

where the softmax temperature τ controls the sampling results between one-hot and uniform. Next, we calculate the hard samples $\mathbf{s}_m$ by $\mathbf{s}_m = \arg\max(\mathbf{z}_m)$. Each row $\mathbf{s}_m$ is derived individually in the same way individually, and the rows are concatenated to form the selection matrix $\mathbf{S}$.

As for gradient estimation, we use the continuous approximation by approximating $\nabla_\theta \mathbf{s} = \nabla_\theta \mathbf{z}$, which is called the Straight-Through (ST) Gumbel Estimator [16]. In summary, by means of the Gumbel-Softmax reparameterisation trick, we can integrate the sampling mechanism for categorical variables into neural networks and train them with standard backpropagation.

B. Algorithm Description and Network Architecture

Since the gradient backpropagation has already been solved, we can complete the DPS algorithm by introducing the loss function. As explained in section I, the loss function can be determined based on the beam pattern, the B-mode image, or the reconstructed image. Therefore, we are interested in finding an optimal selection matrix $\mathbf{S}$ that leads to the best-reconstructed image quality, i.e. minimizes the reconstruction error:

$$\mathcal{L}(\mathbf{S}) = \arg\min_{\mathbf{S}} \|\mathrm{LISTA}(\mathbf{y}_\mathrm{s}) - \mathbf{x}\|_2^2 \tag{5}$$

The neural network consists of two main parts, the pre-trained LISTA for signal recovery, and the Gumbel-Softmax layer for the subsampling matrix. Considering (2) as an example sparse recovery model, the basic ISTA is introduced in Algorithm 1. We rewrite the single ISTA iteration into $\mathbf{x}_{i+1} = h_{\mu\lambda}(\mathbf{W}\mathbf{x}_i + \mathbf{V}\mathbf{y})$ by substituting $\mathbf{W} = \mathbf{I} - \mu\mathbf{A}^\mathrm{T}\mathbf{A}$ and $\mathbf{V} = \mu\mathbf{A}^\mathrm{T}$, where $\mathbf{W}$ and $\mathbf{V}$ can be represented by weights of two separate linear layers. Thus, a LISTA network, as shown in Fig. 2, can be built by replacing the iterations in ISTA with linear layers of a Deep Neural Network (DNN). It is worth noting that the number of layers in a LISTA network can be significantly smaller than the number of iterations in an ISTA/FISTA. For this reason, the LISTA network used in this study is limited to a 20-layer network.

Regarding the Gumbel-Softmax layer, we should first clarify the structure of the selection matrix $\mathbf{S}$. In a simple case, it is preferred not to repeat the same channel in the subsampling

Algorithm 1 Iterative Shrinkage-Thresholding Algorithm

Optimization target: $\arg\min_{\mathbf{x}} \|\mathbf{A}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda\|\mathbf{x}\|_1$

1: Initialization: $\mathbf{x}_1 = \mathbf{0}$, L = Largest singular value of $2\mathbf{A}^\mathrm{T}\mathbf{A}$, step size $\mu = \frac{1}{L}$, $\lambda = 0.1 \cdot \max(\mathbf{A}^\mathrm{T}\mathbf{y})$, $\mathbf{I}$ = Identity matrix.

2: Soft thresholding function:

$$h_{\mu\lambda}(x) = \begin{cases} x - \mu\lambda, & x > \mu\lambda \\ x + \mu\lambda, & x < -\mu\lambda \\ 0, & \text{else} \end{cases}$$

3: **for** $i = 1$ to $N_\mathrm{Iteration}$ **do**

4: $\quad \mathbf{x}_{i+1} = h_{\mu\lambda}((\mathbf{I} - \mu\mathbf{A}^\mathrm{T}\mathbf{A})\mathbf{x}_i + \mu\mathbf{A}^\mathrm{T}\mathbf{y})$

5: **end for**

Output: Recovered signal $\hat{\mathbf{x}}$

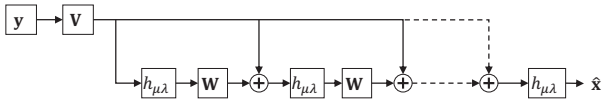

Fig. 2. Deep Neural Network (DNN) architecture of LISTA

design. This means that each column of $\mathbf{S}$ has at most one non-zero element. The usual solution to deal with the matrix is to determine each row separately, and add a penalty term to the loss function. In this case, each row requires at least one layer for representation, but more layers make learning more expensive. To simplify the network architecture, the structure of the selection matrix can be exploited to work instead with the vector $\tilde{\mathbf{s}} \in \mathbb{R}^{N_\mathrm{x}}$ by computing its i-th element as the sum of the i-th column of $\mathbf{S}$. This allows us to optimize the probabilities of M elements in a single vector simultaneously, rather than optimizing each row separately. Next, we replace the matrix multiplication in (3) with an elementwise product to calculate the subsampled data:

$$\mathbf{y}_s = (\tilde{\mathbf{s}} \otimes \mathbf{1}_{N_\mathrm{t}}) \odot \mathbf{y} \tag{6}$$

where $\mathbf{1}_{N_\mathrm{t}} = [1 \quad 1 \ \ldots \ 1]^\mathrm{T} \in \mathbb{R}^{N_\mathrm{t}}$ is a vector of ones. Writing the problem in this way not only allows us to apply a single Gumbel-Softmax layer, but also avoids an additional constraint term in the loss function.

The Fig. 3 is an illustration of the complete network architecture, combining the Gumbel softmax layer and the LISTA network. We assign the categorical probabilities $\boldsymbol{\theta}$ to the biases of a linear layer and set the weights to be zero. Thus, the biases are the only trainable parameters. The reconstruction error is computed by connecting the output sampling vector to the pre-trained LISTA. The detailed computations are explained in the pseudocode Algorithm 2.

C. Data Generation Strategy

We divided the whole specimen map into 16 ROIs indexed by integers $\{1, \ldots, 16\}$ as shown in Fig. 1. According to our prior knowledge, we want to localize and estimate the amplitudes of critical flaws in a particular ROI represented by the sparse vector $\mathbf{x}$ from (2) and (3). The spatial subsampling

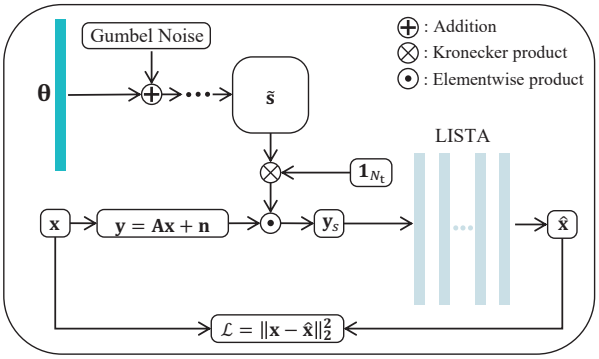

Fig. 3. The training network architecture of Deep Probabilistic Subsampling design. The categorical probability vector $\boldsymbol{\theta}$ is represented by biases of a linear layer with N_{x} inputs and outputs. The pre-trained LISTA network contains 20 layers. Gradient backpropagation is accomplished through the ST Gumbel estimator. Detailed steps in the ellipsis are explained in the following Algorithm 2.

Algorithm 2 Deep Probabilistic Subsampling

Input: Vectorized specimen map $\mathbf{x}$ and randomly generated $\boldsymbol{\theta} \in \mathbb{R}^{N_{\mathrm{x}}}$

Initialization: $M = 24, N_{\mathrm{x}} = 48, N_{\mathrm{z}} = N_{\mathrm{t}} = 256, f_c = 4$ MHz, $f_s = 60$ MHz, $\mathrm{c} = 5850$ m/s, $\Delta\mathrm{x} = 0.5$ mm, $\Delta\mathrm{z} = \mathrm{c}/(2f_s), \tau = 1$
Batch size=5000, $N_{\mathrm{Epochs}} = 20$, step size=10^{-4}
Calculate fully sampled measurement $\mathbf{y}$ by (2)
for $i = 1$ to N_{Epochs} **do**
  Generate i.i.d. Gumbel noise $\boldsymbol{g}_i \in \mathbb{R}^{N_{\mathrm{x}}}$
  Calculate noisy logits: $\mathbf{u}_i = \log \boldsymbol{\theta} + \boldsymbol{g}_i$
  Calculate soft samples (4): $\mathbf{z}_i = \mathrm{softmax}(\mathbf{u}_i)$
  Calculate hard samples $\tilde{\mathbf{s}}_i$
  Calculate subsampled data $\mathbf{y}_s$ by (6)
  Calculate loss: $\mathcal{L}_i = \|\mathrm{LISTA}(\mathbf{y}_{\mathrm{s}}) - \mathbf{x}\|_2^2$
  Compute gradient of the loss
  Use SGD optimizer to update $\boldsymbol{\theta}$
end for

Output: The selection vector $\tilde{\mathbf{s}}$

algorithm should ensure a high-quality reconstructed image if the defects are mainly located in the predefined ROI. In this case, an adatpted data generation strategy aims to improve the sensitivity to detect these critical defects. We assume that the distribution of the point scatterers over the map follows a non-uniform Probability Mass Function (PMF), and that the point scatterers appear with higher probability in a ROI. One can choose how to formulate such a PMF, and Table I shows an example used in our case.

TABLE I
NON-UNIFORM PROBABILITY MASS FUNCTION

	Region of Interest Index				
	10	***11***	***6***	***7***	***others***
Probability Mass	80%	4%	2%	2%	1%

The specific strategy is described as follows. We generate $N_{\mathrm{Training}} = 1,000,000$ training data. The number of point scatterers in each specimen map $\mathbf{x}$ follows the discrete uniform distribution $\mathcal{U}\{1,2,3,4,5\}$. In addition, the scattering amplitude of each scatterer is sampled from the Gaussian distribution $\mathcal{N}(15,3)$. The code is implemented in PyTorch and runs on an NVIDIA A100 GPU node.

IV. EVALUATION

The DPS algorithm has successfully selected $M = 24$ optimal sensor locations from $N_{\mathrm{x}} = 48$. We also include fully sampled and uniformly subsampled grids for comparison, since the fully sampled measurement serves as a baseline reference and the uniform subsampling is the easiest way to implement in simulation or practice. The sensor placement for the three designs above is illustrated in Figure 4.

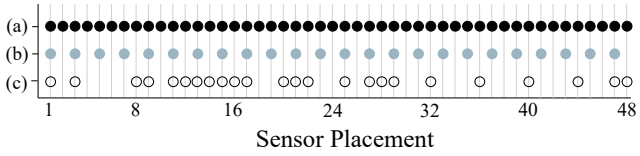

Fig. 4. (a). 48 fully sampled grid; (b). 24 uniformly subsampled grid; (c). 24 DPS grid. The three designs are on the same grid, and only the selection changes.

Following the same strategy, we generate an evaluation dataset of $N_{\mathrm{Evaluation}} = 1000$ data. We then collect the measurement data using the three methods above. To illustrate that the subsampled signal via DPS also works for other SSR algorithms, we use FISTA instead of LISTA in the evaluation. The three procedures are called **Full+FISTA**, **Uni+FISTA** and **DPS+FISTA**. For further quantitative assessment, we introduce two other metrics, Total Squared Error (TSE) and Contrast-to-Noise Ratio (CNR), for the reconstructed signal. The two metrics are reasonable since TSE is related to the signal amplitude and CNR depends on the correctness of the scatterer locations.

A. Total Squared Error (TSE)

The Total Squared Error (TSE) of the reconstructed signal is computed by:

$$\mathrm{TSE} = \|\hat{\mathbf{x}}_{\mathrm{Reconstruction}} - \mathbf{x}_{\mathrm{Groundtruth}}\|_2^2 \tag{7}$$

To visualise the comparison in a simple way, we compute the Cumulative Density Function (CDF) of the $N_{\mathrm{Evaluation}}$ TSEs for the three cases. As shown in Fig. 5, the Uni+FISTA has significantly higher errors and the DPS+FISTA is very close to the Full+FISTA. In terms of reconstruction error, DPS has been shown to be superior to uniform subsampling and slightly inferior to fully sampling.

B. Contrast-to-Noise Ratio (CNR)

The Contrast-to-Noise Ratio (CNR) is defined as:

$$\mathrm{CNR} = \frac{|\mu_i - \mu_o|}{\sqrt{\sigma_i^2 + \sigma_o^2}} \tag{8}$$

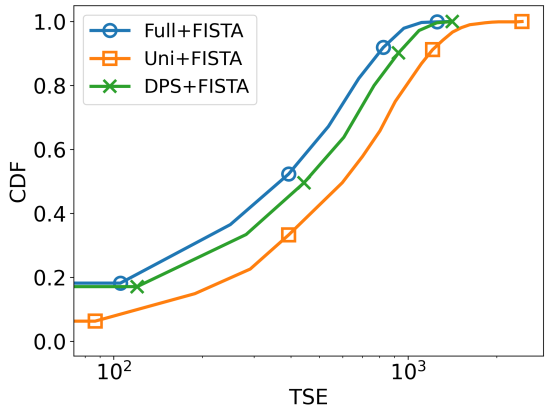

Fig. 5. CDF of TSE for Full+FISTA, Uni+FISTA and DPS+FISTA. A higher and shorter curve means that the majority of results have lower TSE and therefore indicate better performance.

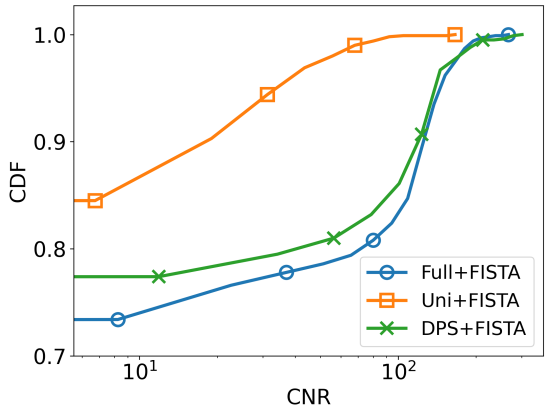

Fig. 6. CDF of CNR for Full+FISTA, Uni+FISTA and DPS+FISTA. A lower and longer curve implies that the majority of results have a higher CNR, indicating better performance.

where μ and σ are the expected value and the variance of the signal power, respectively [17]. The subscripts i and o denote inside and outside the target region. Here, the target region is the true location of each point scatterer.

Observing the following CDF curves versus CNR in Fig. 6, the image quality of Uni+FISTA is obviously worse than the others. Then, from a CNR perspective, DPS has been proven to significantly outperform uniform subsampling and approach fully sampling.

V. Conclusion

This paper demonstrates a model-based data-driven approach within the DPS framework that is able to optimize the subsampling pattern in the field of ultrasound NDT. We contribute to the formulation of the optimal sensor placement problem in single-channel synthetic aperture imaging by constructing an adequate signal model and building an adapted neural network architecture. Meanwhile, we manage to reduce the dimensionality of the desired selection matrix so that the target becomes a vector, which also circumvents an additional penalty term in the cost function, thus speeding up the learning process. For performance evaluation, we assess the TSE and CNR of the reconstructed image obtained via FISTA. Simulations show that DPS significantly outperforms uniformly spatial subsampling and approaches fully sampling. It is worth noting that the value of M is closely linked to the number of flaws and is determined by prior knowledge. Furthermore, our proposed architecture has the potential to be extended to adaptive sensing methods, which can intelligently select the subsequent optimal sensor placement or phased array channel based on previously obtained information during the measurement process.

References

[1] Ramalli A, Boni E, Roux E, et al. Design, implementation, and medical applications of 2-D ultrasound sparse arrays[J]. IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, 2022.

[2] Cohen R, Eldar Y C. Sparse convolutional beamforming for ultrasound imaging[J]. IEEE transactions on ultrasonics, ferroelectrics, and frequency control, 2018, 65(12): 2390-2406.

[3] Diamantaras K, Xu Z, Petropulu A. Sparse antenna array design for MIMO radar using softmax selection[J]. arXiv preprint arXiv:2102.05092, 2021, unpublished.

[4] Wang Z, Bovik A C, Sheikh H R, et al. Image quality assessment: from error visibility to structural similarity[J]. IEEE transactions on image processing, 2004, 13(4): 600-612.

[5] Mamistvalov A, Amar A, Kessler N, et al. Deep-learning based adaptive ultrasound imaging from sub-Nyquist channel data[J]. IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, 2022, 69(5): 1638-1648.

[6] Huijben I A M, Veeling B S, Janse K, et al. Learning sub-sampling and signal recovery with applications in ultrasound imaging[J]. IEEE Transactions on Medical Imaging, 2020, 39(12): 3955-3966.

[7] Pérez E, Kirchhof J, Semper S, et al. Total focusing method with sub-sampling in space and frequency domain for ultrasound NDT[C]//2019 IEEE International Ultrasonics Symposium (IUS). IEEE, 2019: 2103-2106.

[8] Zhang J A, Rahman M L, Wu K, et al. Enabling joint communication and radar sensing in mobile networks—A survey[J]. IEEE Communications Surveys and Tutorials, 2021, 24(1): 306-345.

[9] Demirli R, Saniie J. Model-based estimation of ultrasonic echoes. Part I: Analysis and algorithms[J]. IEEE transactions on ultrasonics, ferroelectrics, and frequency control, 2001, 48(3): 787-802.

[10] Stepinski T. An implementation of synthetic aperture focusing technique in frequency domain[J]. ieee transactions on ultrasonics, ferroelectrics, and frequency control, 2007, 54(7): 1399-1408.

[11] Beck A, Teboulle M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems[J]. SIAM journal on imaging sciences, 2009, 2(1): 183-202.

[12] Gregor K, LeCun Y. Learning fast approximations of sparse coding[C]//Proceedings of the 27th international conference on international conference on machine learning. 2010: 399-406.

[13] Chen X, Liu J, Wang Z, et al. Theoretical linear convergence of unfolded ISTA and its practical weights and thresholds[J]. Advances in Neural Information Processing Systems, 2018, 31.

[14] Jang E, Gu S, Poole B. Categorical reparametrization with gumble-softmax[C]//International Conference on Learning Representations (ICLR 2017). OpenReview. net, 2017.

[15] Maddison C J, Mnih A, Teh Y W. The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables[C]//International Conference on Learning Representations.

[16] Schulman J, Heess N, Weber T, et al. Gradient estimation using stochastic computation graphs[J]. Advances in neural information processing systems, 2015, 28.

[17] Rodriguez-Molares A, Rindal O M H, D'hooge J, et al. The generalized contrast-to-noise ratio[C]//2018 IEEE International Ultrasonics Symposium (IUS). IEEE, 2018: 1-4.