In-Context Learning based Efficient Spectrum Sensing

Renpu Liu
University of Virginia
pzw7bx@virginia.edu

Liwen Zhong
Pennsylvania State University
lbz5228@psu.edu

Wooram Lee
Pennsylvania State University
wbl5187@psu.edu

Jing Yang
University of Virginia
yangjing@virginia.edu

***Abstract*—The radio frequency (RF) spectrum is essential for wireless communication but is becoming increasingly limited due to the rapid growth in device usage. Real-time spectrum sensing facilitates dynamic spectrum sharing, but conventional methods face significant challenges, including the high power consumption of analog-to-digital converters (ADCs) and the computational demands of Fast Fourier Transforms (FFTs). To address these limitations, prior work introduced a frequency-domain analog signal processor. This processor includes a digitally tunable narrow-bandpass filter implemented with programmable dispersion-engineered elements and a scalable path-sharing delayed signal combiner. However, the naive spectrum sweeping method employed in this design remains highly time- and energy-intensive.**

In this work, we improve the spectrum sensing efficiency of the analog signal processor by co-designing a sensing matrix generation method with a decoder-based transformer for in-context spectrum recovery. Specifically, we introduce a novel algorithm for sensing matrix generation that leverages the hardware design of the analog signal processor. We show that the generated sensing matrices can be interpreted as part of the well-designed prompts for a transformer with specifically designed parameter matrices to solve the sparse spectrum sensing problem efficiently through its in-context learning capability.

To characterize the efficiency of the in-context learning-enabled spectrum sensing approach, we provide rigorous theoretical guarantees on the in-context spectrum sensing and evaluate the performances through empirical results. Compared to baseline approaches, our method achieves significant improvements in accuracy.

I. Introduction

As emerging applications in areas such as 5G communications and satellite links migrate toward millimeter-wave (mmWave) frequencies, the need for broadband spectrum sensing that can cover wide-range bands is becoming urgent. However, implementing real-time wideband scanning using conventional digital techniques presents fundamental challenges. These include the excessive power consumption and complexity of high-speed analog-to-digital converters (ADCs) and digital signal processing (DSP) circuits required to perform large-scale Fast Fourier Transforms (FFTs).

To address these challenges, a frequency-domain analog processor has been recently proposed [1], offering an energy-efficient solution for wideband spectrum sensing. This processor draws inspiration from frequency-scanned arrays and leverages programmable dispersion-engineered elements to exploit frequency-dependent constructive and destructive interference. At its core, the design incorporates an array of programmable dispersive phase shifters, all sharing a common RF input.

Meanwhile, transformers have become the backbone of various machine learning tasks, including natural language processing [2], [3] and computer vision [4], [5], significantly influencing subsequent research and applications. A key strength of transformers is their strong performance in in-context learning (ICL) [6], where they can perform new inference tasks based on the contextual information embedded in example input-output pairs provided in the prompt, without requiring further parameter updates.

In this work, we leverage the ICL capabilities of transformers to enhance the spectrum sensing efficiency of the analog processor. Specifically, we demonstrate that a pre-trained transformer can be deployed with the analog processor to perform real-time spectrum sensing with provable performance guarantees.

Our main contributions are as follows:

- First, we propose a sensing matrix generation algorithm, together with a transformer parameter implementation, to enable the transformer's in-context sparse spectrum sensing. The sensing matrix generation algorithm produces sensing matrices, which will be combined with the corresponding output power of the analog signal processor to form the prompt for the transformer. The sensing matrix generation algorithm is specifically designed to enhance the transformer's capacity for solving in-context spectrum sensing problems. Additionally, the parameters of the transformer are optimized to efficiently utilize the generated prompts (sensing matrices) for ICL.
- Second, we theoretically characterize the performance of the proposed in-context spectrum sensing approach. More specifically, we show that when the number of measurements is of the order $\mathcal{O}(S^2 \log L)$, where S is the sparsity of the power spectrum vector and L is the number of subbands, the estimated power spectrum produced by the transformer converges to the ground truth at a rate linear in the depth of the transformer model. Such results indicate that the proposed in-context spectrum sensing can significantly reduces sensing time and energy consumption.
- Third, we perform simulation experiments to validate the efficiency of our algorithm and support our theoreti-

This work was partially supported by the US National Science Foundation under Grant ECCS-2318759.

cal findings. The results demonstrate that the in-context spectrum sensing method can achieve more accurate estimation compared to classic baseline algorithms.

II. Preliminaries and Problem Formulation

Notation. Bold uppercase letters (e.g., $\mathbf{X}$) denote matrices, and bold lowercase letters (e.g., $\mathbf{x}$) denote vectors. The ℓ_p norm of a vector $\mathbf{x}$ is denoted by $\|\mathbf{x}\|_p$. The Hadamard (element-wise) product is denoted by $\odot$. The identity matrix of size $d\times d$ is denoted by $\mathbf{I}_d$, while $\mathbb{1}_{M\times N}$ represents the all-1 matrix with dimensions $M \times N$. The conjugate of a complex number $x \in \mathbb{C}$ is denoted by $\bar{x}$.

A. Frequency Domain Analog Processor

The algorithm design is abased on a recently proposed frequency-domain analog signal processor [1] as depicted in Figure 1(a). It consists of an array of N programmable dispersion-engineered elements that share a common RF input. Each dispersion-engineered element, labeled as k, $k \in [0 : N-1]$, is implemented with the combination of a broadband phase shifter, which exhibits constant phase shift $k\Delta\phi$ across frequency, and a fixed true-time delay line with delay $k\Delta\tau$. Therefore, for incoming signal at frequency ω, it will encounter a phase shift at each element k by $\theta_k(\omega) = k(\Delta\phi - \Delta\tau\cdot\omega)$.

Let a_k represent the weight of each element, which is controlled via the gain setting of the phase shifters. If $a_k = 0$, element k is effectively turned off. Then, the array factor (AF), defined as the frequency response of an N-element array, is given by

$$AF(\omega) = \sum_{k=0}^{N-1} a_k \cdot e^{j\theta_k(\omega)}. \tag{1}$$

$$AF(\omega) = \sum_{k=0}^{M-1} a_k \cdot e^{jk(\Delta\phi-\Delta\tau\cdot\omega)}. \tag{2}$$

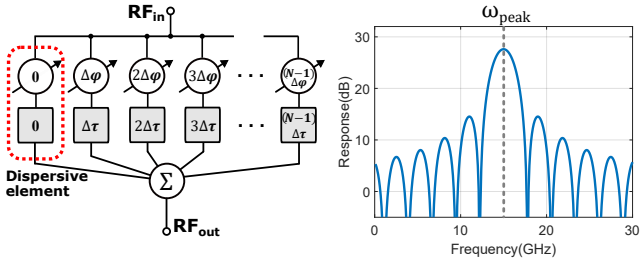
Figure 1(b) shows the calculated frequency response of the proposed spectrum sensor for a specific $\Delta\phi$, $\Delta\tau$, and identical a_k for all k. The response is similar to a band-pass filter, where the maximum array factor (AF) is achieved when the signals are combined in phase.

Let $P(\omega)$ be the power spectrum density (PSD) function of the input signal $x(t)$. Then, the power of the output signal of the analog processor equals $\int_\omega P(\omega)|AF(\omega)|^2 d\omega$, which is measured by an RF power detector.

Intuitively, by adjusting the analog processor configuration $\{\Delta\phi, \Delta\tau, \{a_k\}_{k=0}^{N-1}\}$ and measuring the corresponding output power, the analog processor can obtain "sketches" of the spectrum occupation, which can then be used to reconstruct the entire spectrum.

B. Spectrum Estimation as Constrained Sparse Recovery

We cast the spectrum sensing problem into a sparse recovery problem as follows. We assume the spectrum can be discretized uniformly into L subbands, where the central frequency of the ℓ-th subband is ω_l. We assume there



(a) (b)

Fig. 1: (a) Proposed frequency-domain analog processor. (b) Simulated frequency response.

exists a w' such that $w_\ell = \ell \cdot w'$ for all $\ell \in [L]$. We assume $P(\omega)$ can be approximated as a constant p_ℓ in each subband, and the corresponding $AF(\omega)$ can be approximated as $AF(\omega_\ell)$. Let $\mathbf{p} = [p_1, p_2, \ldots, p_L]^T$, and $\mathbf{h} = [AF(\omega_1), AF(\omega_2), \ldots, AF(\omega_L)]^T$. Then, the measured output power can be approximated as $y = (\mathbf{h}\odot\overline{\mathbf{h}})^\top\mathbf{p} + \epsilon$, where ϵ represents the measurement noise.

Note that $\mathbf{h}$ depends on the chosen configuration $\{\Delta\phi, \Delta\tau, \{a_k\}_{k=1}^M\}$. Assume we have M such configurations, and correspondingly, we obtain $\{y_i\}_{i=1}^M$, where $y_i = (\mathbf{h}_i \odot \overline{\mathbf{h}}_i)^\top\mathbf{p} + \epsilon_i$. Assume there are at most $S \ll L$ subbands are occupied. Then, the spectrum estimation problem can be formulated as a sparse recovery problem:

$$P_1: \quad \min_{\mathbf{p}\in\mathbb{R}^L} \frac{1}{M}\sum_{i=1}^{M}\left((\mathbf{h}_i\odot\overline{\mathbf{h}}_i)^\top\mathbf{p} - y_i\right)^2, \quad \text{s.t. } \|\mathbf{p}\|_0 \le S.$$

We denote the optimizer of P_1 as $\mathbf{p}^\star$. Since solving P_1 is NP-hard, a common relaxation is to consider the Least Absolute Shrinkage and Selection Operator (LASSO) formulation:

$$\min_{\mathbf{p}\in\mathbb{R}^L} \frac{1}{M}\sum_{i=1}^{M}\left((\mathbf{h}_i\odot\overline{\mathbf{h}}_i)^\top\mathbf{p} - y_i\right)^2 + \lambda\|\mathbf{p}\|_1.$$

Let $\widetilde{\mathbf{h}}_i = \mathbf{h}_i \odot \overline{\mathbf{h}}_i$ and

$$\mathbf{H} = \begin{bmatrix} \widetilde{\mathbf{h}}_i^\top \\ \vdots \\ \widetilde{\mathbf{h}}_i^\top \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix}.$$

Then, the LASSO formulation can be expressed as

$$P_2: \quad \min_{\mathbf{p}\in\mathbb{R}^L} \|\mathbf{H}\mathbf{p} - \mathbf{y}\|^2 + \lambda\|\mathbf{p}\|_1. \tag{3}$$

Extensive research has been performed in the past years to solve P_2 reliably and efficiently. For sensing matrices $\mathbf{H}$, random designs (e.g., Gaussian, sub-Gaussian, Bernoulli) are widely used due to their strong recovery guarantees, while structured designs (e.g., Fourier transforms, expander graphs) are often employed for specific applications [7]. Existing results show that when the number of measurements is $\mathcal{O}(S\text{poly}\log L)$, random matrices have strong recovery guarantees typically based on properties like the Restricted Isometry Property (RIP) and Null Space Property [8]–[10]. While for structured designs such as partial Fourier matrices, theoretical studies show that if one forms $\mathbf{H}$ by selecting m

rows of a discrete Fourier matrix uniformly, then $\mathbf{H}$ satisfies RIP with high probability when $m = \mathcal{O}(S\text{poly}\log L)$ [9], [11].

Meanwhile, iterative algorithms, such as ISTA and LISTA, are shown to achieve sublinear convergence rates [12]. Alternatively, learning-to-optimize (L2O) based solvers, such as LISTA [13], LISTA-CP [14], and ALISTA [15], can achieve linear convergence rates. However, these L2O-based solvers require retraining the solver model for each new sensing matrix, which introduces additional computational overhead.

C. Transformer and In-context Learning

We consider a K-layer decoder-based transformer architecture [16], where each transformer layer has an attention layer masked by a decoder-based attention mask and followed by a multi-layer perception (MLP) layer.

Masked Attention Layer. An P-head masked attention layer with parameters $\{(\mathbf{V}_m, \mathbf{Q}_m, \mathbf{K}_m)_{m\in[P]}\}$ where $\mathbf{V}_m, \mathbf{Q}_m, \mathbf{K}_m \in \mathbb{R}^{D\times D}$, is denoted as $\text{Attn}_{\{(\mathbf{V}_m,\mathbf{Q}_m,\mathbf{K}_m)\}}(\cdot)$. For an input sequence $\mathbf{Z} \in \mathbb{R}^{D\times N}$, its output is:

$$\begin{aligned}&\text{Attn}_{\{(\mathbf{V}_m,\mathbf{Q}_m,\mathbf{K}_m)\}}(\mathbf{Z}) \\ &= \mathbf{Z} + \sum_{m=1}^{P} (\mathbf{V}_m\mathbf{Z}) \times \text{mask}\Big(\sigma\big((\mathbf{K}_m\mathbf{Z})^\top(\mathbf{Q}_m\mathbf{Z})\big)\Big),\end{aligned} \tag{4}$$

where $\text{mask}(\mathbf{M})$ satisfies $\text{mask}(\mathbf{M})_{i,j} = \frac{1}{j}\mathbf{M}_{i,j}$ if $i \le j$ and 0 otherwise, and $\sigma(\cdot)$ is the activation function. In this work, $\sigma(\cdot)$ is set as ReLU.

MLP Layer. An MLP layer with parameters $\mathbf{W}_1 \in \mathbb{R}^{D'\times D}$, $\mathbf{W}_2 \in \mathbb{R}^{D\times D'}$, and $\mathbf{b} \in \mathbb{R}^{D'}$, denoted as $\text{MLP}_{\{\mathbf{W}_1,\mathbf{W}_2,\mathbf{b}\}}$. Denote the output sequence of attention layer as $\mathbf{Z}' = \text{Attn}(\mathbf{Z})$, the MLP layer maps each column $\mathbf{z}'_i$ of $\mathbf{Z}' \in \mathbb{R}^{D\times N}$ as:

$$\text{MLP}_{\{\mathbf{W}_1,\mathbf{W}_2,\mathbf{b}\}}(\mathbf{z}'_i) = \mathbf{z}'_i + \mathbf{W}_2\sigma(\mathbf{W}_1\mathbf{z}'_i + \mathbf{b}). \tag{5}$$

In-Context Learning (ICL). For an ICL task, a trained transformer is given an ICL instance $\mathcal{I} = (\mathcal{D}, \widetilde{\mathbf{h}}_{M+1})$, where $\mathcal{D} = \{(\widetilde{\mathbf{h}}_i, y_i)\}_{i\in[M]}$ and $\widetilde{\mathbf{h}}_{M+1}$ is a query. Here, $\widetilde{\mathbf{h}}_i \in \mathbb{R}^d$ is an in-context example, and y_i is the corresponding label for $\widetilde{\mathbf{h}}_i$. We assume $y_i = f_{\mathbf{p}}(\widetilde{\mathbf{h}}_i) + \epsilon_i$, where ϵ_i is an added random noise, and $f_{\mathbf{p}}$ is a deterministic function parameterized by $\mathbf{p}$. Unlike conventional supervised learning, for each ICL instance, $\mathbf{p} \sim P_{\mathbf{p}}$, i.e., it is randomly sampled from a distribution $P_{\mathbf{p}}$.

To perform ICL in a transformer, we first embed the ICL instance into an input sequence $\mathbf{Z} \in \mathbb{R}^{D\times M'}$. The transformer then generates an output sequence $\text{TF}(\mathbf{Z})$ with the same size as $\mathbf{Z}$, based on which a prediction $\widehat{y}_{M+1}$ is generated through a read-out function F, i.e., $\widehat{y}_{M+1} = F(\text{TF}(\mathbf{Z}))$. The objective of ICL is then to ensure that $\widehat{y}_{M+1}$ closely approximates the target value $y_{M+1} = f_{\mathbf{p}}(\widetilde{\mathbf{h}}_{M+1}) + \epsilon_{M+1}$ for any ICL instance.

During the pre-training of a transformer for an ICL task, it first samples a large set of ICL instances. For each instance, the transformer generates a prediction $\widehat{y}_{M+1}$ and calculates the prediction loss by comparing it with y_{M+1} using a proper loss function. The training loss is the aggregation of all prediction losses for every ICL instance used in pre-training, and the transformer is trained to minimize this training loss.

D. In-context Learning for Spectrum Estimation

Our objective is to leverage the ICL capability of transformers to perform efficient spectrum sensing.

During the pre-training process, a set of in-context spectrum sensing instances $\{(\mathbf{H}^j, \mathbf{y}^j, \widetilde{\mathbf{h}}^j_{M+1}, y^j_{M+1})\}_{j=1}^{N_{\text{train}}}$ is generated according to

$$y^j_i = (\widetilde{\mathbf{h}}^j_i)^\top \boldsymbol{p}^j + \epsilon^j_i, \quad j \in [N_{\text{train}}],\, i \in [M+1],$$

where $\boldsymbol{p}^j \sim P_{\mathbf{p}}$, $\mathbf{h}^j_i \sim P_{\mathbf{h}}$, and $\epsilon^j_i \sim P_\epsilon$ are independently sampled from their respective distributions. Based on this training dataset, a pre-trained transformer is obtained by minimizing a specified loss function.

After pre-training, during the inference process for ICL, a spectrum sensing instance $(\mathbf{H}, \mathbf{y}, \widetilde{\mathbf{h}}_{M+1})$ is randomly sampled according to the same distributions as in the pre-training. *Different from conventional ICL where the objective is to predict y_{M+1}, our objective is to explicitly estimate the hidden $\mathbf{p}$ based on the input $(\mathbf{H}, \mathbf{y}, \widetilde{\mathbf{h}}_{M+1})$ without any further parameter updates.* As we will show in the subsequent section, this can be achieved through a joint design of the configuration of the analog processor and the transformer.

III. Joint Design and Analysis

To ensure the sensing matrices given by the analog processor can theoretically guarantee a transformer can recover sparse frequency spectrum in-context, we need to co-design both the sensing matrix generation method and the transformer implementation, i.e., $\{\mathbf{V}_m, \mathbf{Q}_m, \mathbf{K}_m\}$ in the attention layers and $\{\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}\}$ in the MLP layers.

A. Configuration of the Analog Processor

We start with the design of the configuration of the analog processor to obtain the sensing matrices. In the literature on compressive sensing, particularly those focused on the theoretical performance analysis of compressive sensing algorithms, it often assumes random sensing matrices, typically with i.i.d. Gaussian entries. However, in the context of the proposed analog processor-based spectrum estimation, the sensing matrix $\mathbf{H}$ is inherently constrained by the array configurations, which are determined by the physical properties of the system, including structural and power limitations. Consequently, rather than relying on generic random matrices, we focus on designing structured, non-adaptive sensing schemes by strategically controlling $\Delta\phi$, $\Delta\tau$ and $\{a_k\}_k$ to enable efficient and practical spectrum estimation. Our method consists of four major steps:

a) Random Tuple Selection: From the set $\{1, 2, \ldots, N-1\}$, randomly select an M-element tuple $\mathcal{I} = [\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_M]$. Each element $\mathcal{I}_i$ represents a spacing parameter that will guide the selection of element pairs.

b) Random Pair Selection: For each $i \in [M]$, randomly select a pair of element indices $\{k_{i,1}, k_{i,2}\}$ such that $|k_{i,1} - k_{i,2}| = \mathcal{I}_i$. This pair corresponds to the array elements that will be activated for the i-th measurement configuration.

Algorithm 1 Random Pair Activation

INPUT N, M.
Randomly select a tuple $\mathcal{I} = [\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_M]$ with distinct elements of size M from $\{1, 2, \ldots, N-1\}$.
for $i = 1$ to M **do**
  Randomly sample an index pair $\{k_{i,1}, k_{i,2}\}$ where $k_{i,1}, k_{i,2} \in \{0, \cdots, N-1\}$ such that $k_{i,1} \neq k_{i,2}$ and $|k_{i,1} - k_{i,2}| = \mathcal{I}_i$.
  for each element index k **do**
    if $k \in \{k_{i,1}, k_{i,2}\}$ **then**
      $a_k^{(i)} \leftarrow 1$
    else
      $a_k^{(i)} \leftarrow 0$
    end if
  end for
  $\Delta\phi^{(i)} \leftarrow \frac{\pi}{2N}$, $\Delta\tau^{(i)} \leftarrow \frac{\pi}{\omega' N}$ for all $i \in [M]$.
end for

c) Element Activation: We set $a_k^{(i)} = 1$ if $k \in \{k_{i,1}, k_{i,2}\}$, and $a_k^{(i)} = 0$ otherwise. Each configuration thus activates exactly two elements, reducing power consumption while still achieving sufficient measurement diversity.

d) Time-Delay and Phase Control: We set the time-delay and phase as $\Delta\tau^{(i)} = \frac{\pi}{\omega' N}$ and $\Delta\phi^{(i)} = \frac{\pi}{2N}$.
$\Delta\tau^{(i)} = \frac{\pi}{\omega' M}$, $\Delta\phi^{(i)} = \frac{\pi}{2M}$.

B. Transformer Design

We set the number of heads P to 4, therefore the attention layer contains four parameter matrices $\{\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i\}$, $i \in \{1, -1, 2, -2\}$. We set these parameter matrices as follows:

$$\mathbf{Q}_i = \mathrm{diag}(\mathbf{0}_{(L+1)\times(L+1)}, i\mathbf{I}_{L\times L}, -B), \text{ for } i \in \{\pm 1\},$$
$$\mathbf{Q}_i[L+1, 2L+2] = i/2, \text{ for } i \in \{\pm 1\},$$
$$\mathbf{V}_i[L+2 : 2L+1, 1 : L] = \mathrm{sgn}(i)\gamma\mathbf{I}_{L\times L}, \text{ for } i \in \{\pm 1, \pm 2\},$$

where sgn is the sign function and B, γ are constants. The parameter implementation for $\mathbf{K}_i$ matrices follows the setup described in Appendix C.1 of [17].

We note that the design of the transformer resembles that in [17] for the general sparse recovery problem. However, there exists a critical difference: in this work, all parameter matrices in the attention layer are fixed, meaning they do not contain any learnable parameters. Consequently, this fixed transformer does not require learning those parameter matrices from pre-training. This is because all sensing matrices generated by Algorithm 1 possess the same statistical properties, such as the concentration property of mutual coherence, which can be calculated beforehand. As a result, these statistical properties do not need to be learned from pre-training.

For the input data, we adopt a similar embedding structure as in [17], [18]. Given an in-context sparse recovery instance $\mathcal{I} = (\mathbf{H}, \mathbf{y})$, we embed the instance into an input sequence $\mathbf{Z}^{(1)} \in \mathbb{R}^{(2L+2)\times(2N+1)}$ as follows:

$$\mathbf{Z}^{(1)}(\mathcal{I}) = \begin{bmatrix} \widetilde{\mathbf{h}}_1 & \widetilde{\mathbf{h}}_1 & \cdots & \widetilde{\mathbf{h}}_M & \widetilde{\mathbf{h}}_M \\ 0 & y_1 & \cdots & 0 & y_N \\ \widehat{\mathbf{p}}_1^{(1)} & \widehat{\mathbf{p}}_2^{(1)} & \cdots & \widehat{\mathbf{p}}_{2M-1}^{(1)} & \widehat{\mathbf{p}}_{2M}^{(1)} \\ 1 & 0 & \cdots & 1 & 0 \end{bmatrix},$$

where $\mathbf{p}_i^{(1)} \in \mathbb{R}^L$ are implicit parameter vectors initialized as $\mathbf{0}_d$, and $\mathbf{h}_i$ is the i-th sensing vector, the superscript (1) denotes that it is the input sequence for the first layer.

Denote the output sequence of the K-layer Transformer as $\mathbf{Z}^{(K+1)}$. During the inference, we obtain the estimated spectrum $\widehat{\mathbf{p}}_{2M}^{(K+1)}$ from $\mathbf{Z}^{(K+1)}[L+2 : 2L+1, 2M]$.

C. Theoretical Guarantee

In this section, we present a theoretical result that provides a rigorous guarantee for using the transformer implementation described in Section III-B and the sensing matrix generation method outlined in Section III to solve the in-context spectrum recovery problem. To establish this result, we first introduce the following assumption, which has been adopted in similar forms in recent works [17], [18].

Assumption 1. *For each input signal, the total power is bounded, i.e., there exists a positive constant E such that $\|\mathbf{p}\|_1 \leq E$.*

The following theorem provides a rigorous theoretical guarantee for the performance of our proposed sensing scheme. In essence, it shows that if we choose sufficiently many measurement configurations, the resulting sensing matrix $\mathbf{H}$ will, with high probability, preserve the structure of all sufficiently sparse signals. This property ensures that the minimizer $\mathbf{p}^*$ of P_1 closely approximates the true sparse vector that we seek to recover.

Theorem 1. *Let $\delta \in (0,1)$, $M \geq c_1 S^2(\log L + \log S - \log\delta)$, $\alpha = -\log\left(1 - \frac{2}{3}\gamma + \gamma(2S-1)\sqrt{\frac{\log d - \log\delta}{c_2 M}} + \sqrt{\frac{\log S - \log\delta}{c_2 M}}\right)$, where c_1, c_2 and γ are positive constants with $\gamma \leq \frac{3}{2}$. For a K-layer transformer model specified in Section III-B, with any randomly generated $\mathbf{p}$ satisfying Assumption 1 and any sensing matrix generated by Algorithm 1, in the noiseless case, with probability at least $1-\delta$, we have $\left\|\widehat{\mathbf{p}}_{2M}^{(K+1)} - \mathbf{p}\right\| \leq Ee^{-\alpha_n K}$.*

Proof Sketch. The proof of Theorem 1 proceeds in two steps.

Step 1 (Connection to DCT): We begin by linking the sensing matrix generated by Algorithm 1 to the Discrete Cosine Transform (DCT) matrix. By substituting the chosen parameters $\{\Delta\tau_i, \Delta\phi_i\}_i$ from Algorithm 1, we have

$$\mathbf{h}_i = \underbrace{\begin{bmatrix} 1 & e^{-j\frac{\pi}{2N}} & \cdots & e^{-j(N-1)\frac{\pi}{2N}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j\frac{\pi}{N}(L-\frac{1}{2})} & \cdots & e^{-j\frac{\pi}{N}(N-1)(L-\frac{1}{2})} \end{bmatrix}}_{\mathbf{W}_N} \begin{bmatrix} a_{0i} \\ \vdots \\ a_{(N-1)i} \end{bmatrix}.$$

Since $\frac{1}{\sqrt{N}}\mathbf{W}_N$ is closely related to the block Discrete Fourier Transform (DFT) matrix, we can establish a direct

connection between $\mathbf{h}_i$ and DCT basis. Recall that $\mathbf{H} = [\widetilde{\mathbf{h}}_1 \cdots \widetilde{\mathbf{h}}_M] = [\mathbf{h}_1 \odot \overline{\mathbf{h}}_1 \cdots \mathbf{h}_M \odot \overline{\mathbf{h}}_M]$, we obtain the (i,ℓ)-th element of $\mathbf{H}$ as

$$\mathbf{H}_{i,\ell} = 2 + 2\cos\left(\pi \frac{\ell - \frac{1}{2}}{N}(k_{i,1} - k_{i,2})\right),$$

$$x_i[\ell] = 2 + 2\cos\left(\pi \frac{\ell - \frac{1}{2}}{N}(k_{i,1} - k_{i,2})\right),$$

where $i, \ell \in \{1, \cdots, N\}$. From this, we note that $\frac{1}{\sqrt{2N}}(\mathbf{H} - \mathbb{2}_{M\times L})$ can be seen as generated from the following procedure: first randomly select M rows from a DCT matrix $\mathbf{C}_N$ (except the first row), and then select the submatrix that only contains the first L columns of the randomly generated matrix, where the DCT matrix $\mathbf{C}_N$ is:

$$\mathbf{C}_N = \begin{bmatrix} \frac{1}{\sqrt{N}} & \cdots & \frac{1}{\sqrt{N}} \\ \vdots & \ddots & \vdots \\ \sqrt{\frac{2}{N}}\cos\left(\frac{\pi(N-1)}{2K}\right) & \cdots & \sqrt{\frac{2}{N}}\cos\left(\frac{\pi(N-\frac{1}{2})(N-1)}{N}\right) \end{bmatrix}.$$

Step 2 (Linear convergence): Note that by constructing the MLP layer, we show that the updating rule of the transformer is equal to

$$\widehat{\mathbf{p}}_{2M}^{(k+1)} = \mathcal{S}_{\theta^{(k)}}\left(\widehat{\mathbf{p}}_{2M}^{(k)} - \frac{1}{2M}\widetilde{\mathbf{H}}^\top(\widetilde{\mathbf{H}}\widehat{\mathbf{p}}_{2M}^{(k)} - \widetilde{\mathbf{y}})\right), \quad (6)$$

where $\widetilde{\mathbf{H}} = \frac{1}{\sqrt{2N}}(\mathbf{H} - \mathbb{2}_{M\times L})$ and $\widetilde{\mathbf{y}} = \frac{\mathbf{y}}{\sqrt{2N}} - \frac{E\mathbb{2}_K}{\sqrt{2N}}$. Utilizing the orthonormal property of DCT matrix, similar to the proof in compressive sensing [9], [10] showing the RIP of such matrix, we show that for any sensing matrix $\mathbf{H}$ generated by Algorithm 1, $\frac{1}{\sqrt{2N}}(\mathbf{H} - \mathbb{2}_{M\times L})$ satisfies the mutual coherence requirement given in Lemma 3 in [17] with high probability. Therefore, utilizing Lemma 1 in [17] we can show the updating rule in Equation (6) gives linear convergence rate.

Remark 1. *In Theorem 1, the number of measurements M is of the order $\mathcal{O}(S^2 \log L)$. Moreover, it can be shown that this M ensures the sensing matrix generated by Algorithm 1 satisfies $RIP(2S, \delta)$ for a fixed δ with high probability. This guarantees that our sensing matrix generation method produces matrices capable of accurate sparse recovery.*

Remark 2. *Theorem 1 shows that the number of measurements M grows logarithmically with respect to L. Consequently, when S is small, the number of measurements required to accurately recover the sparse spectrum is $\mathcal{O}(S^2 \log L)$, which significantly reduces the number of measurements compared to the naive spectrum sweeping method that requires $\mathcal{O}(L)$ measurements.*

IV. Experimental Results

In our experiments, we adhere to the following steps to generate in-context spectrum sensing instances. First, we set $\mathbf{p}^*$ as an $L = 20$ dimensional vector with the sparsity 3, where 3 entries are randomly chosen and set to be 1. Next, we generate two types of sensing matrices of $\mathbf{H}$ of dimension 10×20: the first type of sensing matrices is generated from Algorithm 1, and the second type of sensing matrices is generated by random sampling each entry from a standard normal distribution. We follow the noiseless setting in [17], [18] for sparse recovery, i.e., $\mathbf{y} = \mathbf{H}\mathbf{p}^*$.

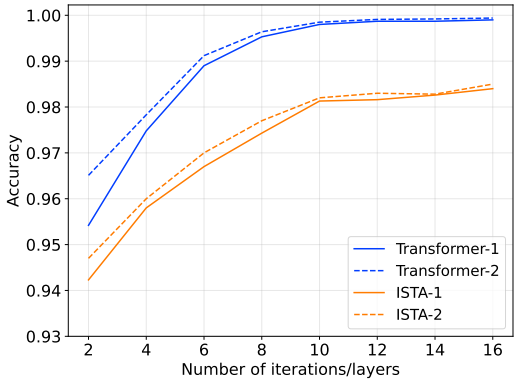


Fig. 2: Accuracy of the in-context spectrum recovery problem for different sensing matrix generation methods and solvers.

We consider two types of solvers for solving the in-context spectrum sensing problem. The first is the transformer with the implementation introduced in Section III-B, and the other is the ISTA algorithm [12]. Figure 2 shows the test accuracy of the in-context spectrum recovery problem for different sensing matrix generation methods and solvers. The accuracy is defined as $|\widehat{I} \cap I^\star|/S$, where $\widehat{I}$ and $I^\star$ represent the supports of the prediction $\widehat{\mathbf{p}}$ and the ground truth $\mathbf{p}^\star$, respectively.

In Figure 2, "Transformer-1" refers to using the transformer as the solver with sensing matrices generated by Algorithm 1, while "Transformer-2" refers to sensing matrices generated from a standard Gaussian distribution. Similarly, "ISTA-1" and "ISTA-2" indicate ISTA solvers with the same respective sensing matrix generation methods.

Our results indicate that for both solvers, sensing matrices generated by Algorithm 1 achieve slightly degraded performance compared to standard Gaussian random matrices. However, the results also show that the transformer recovers the sparse spectrum with higher accuracy than ISTA under the same number of layers (for the transformer) or iterations (for ISTA), showcasing the efficiency of in-context learning (ICL) for spectrum estimation. Additionally, our experimental results demonstrate that when the number of transformer layers is large (greater than 10), our co-designed sensing matrix and transformer implementation provide relatively accurate estimation with significantly fewer measurements ($M = 10$) compared to the naive spectrum sweeping method, which requires $L = 20$ measurements.

V. Conclusion

In this work, we investigated the joint design of the configuration of an analog processor and a transformer to enable efficient in-context spectrum sensing. We theoretically characterized the superb performance of the proposed in-context spectrum sensing method, and validated it through simulations. Compared with traditional sublinear iterative algorithms, the proposed in-context spectrum sensing approach achieves linear

convergence rate without retraining the transformer for different sensing matrices.

REFERENCES

[1] L. Zhong, M. Abbasi, S. M. A. Uddin, and W. Lee, "Broadband frequency-domain analog processor for spectrum sensing with20 ghz scan range," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 5, pp. 1759–1763, 2023.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 2019, pp. 4171–4186.

[3] A. Liu, B. Feng, B. Xue, *et al.*, "Deepseek-v3 technical report," *arXiv preprint arXiv:2412.19437*, 2024.

[4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations*, 2020.

[5] B. Wu, C. Xu, X. Dai, *et al.*, "Visual transformers: Token-based image representation and processing for computer vision," *arXiv preprint arXiv:2006.03677*, 2020.

[6] T. Brown, B. Mann, N. Ryder, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[7] M. Lustig, D. Donoho, and J. P. S. MRI, "Application of" compressed sensing" for rapid mr imaging (2005)," *DOI: https://doi. org/10.1002/mrm*, vol. 21391, pp. 1182–1195,

[8] D. L. Donoho, "Compressed sensing," *IEEE Transactions on information theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[9] M. Rudelson and R. Vershynin, "On sparse reconstruction from fourier and gaussian measurements," *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 61, no. 8, pp. 1025–1045, 2008.

[10] K. Li, L. Gan, and C. Ling, "Convolutional compressed sensing using deterministic sequences," *IEEE Transactions on Signal Processing*, vol. 61, no. 3, pp. 740–752, 2012.

[11] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on information theory*, vol. 52, no. 2, pp. 489–509, 2006.

[12] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM journal on imaging sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[13] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proceedings of the 27th international conference on international conference on machine learning*, 2010, pp. 399–406.

[14] X. Chen, J. Liu, Z. Wang, and W. Yin, "Theoretical linear convergence of unfolded ista and its practical weights and thresholds," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[15] J. Liu and X. Chen, "Alista: Analytic weights are as good as learned weights in lista," in *International Conference on Learning Representations (ICLR)*, 2019.

[16] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[17] R. Liu, R. Zhou, C. Shen, and J. Yang, "On the learn-to-optimize capabilities of transformers in in-context sparse recovery," in *International Conference in Learning Representation (ICLR)*, 2025.

[18] Y. Bai, F. Chen, H. Wang, C. Xiong, and S. Mei, "Transformers as statisticians: Provable in-context learning with in-context algorithm selection," *Advances in neural information processing systems*, vol. 36, 2024.