Decision Feedback In-Context Symbol Detection over Block-Fading Channels

Li Fan, Jing Yang, Cong Shen
Charles L. Brown Department of Electrical and Computer Engineering, University of Virginia, USA
E-mails: {lf2by,yangjing,cong}@virginia.edu

***Abstract*—Pre-trained Transformers, through in-context learning (ICL), have demonstrated exceptional capabilities to adapt to new tasks using example prompts *without model update*. Transformer-based wireless receivers, where prompts consist of the pilot data in the form of transmitted and received signal pairs, have shown high estimation accuracy when pilot data are abundant. However, pilot information is often costly and limited in practice. In this work, we propose the DEcision Feedback IN-ContExt Detection (`DEFINED`) solution as a new wireless receiver design, which bypasses channel estimation and directly performs symbol detection using the (sometimes extremely) limited pilot data. The key innovation in `DEFINED` is the proposed decision feedback mechanism in ICL, where we sequentially incorporate the detected symbols into the prompts to improve the detections for subsequent symbols. Extensive experiments across a broad range of wireless communication settings demonstrate that `DEFINED` achieves significant performance improvements, in some cases only needing a single pilot pair.**

I. Introduction

Wireless receiver symbol detection focuses on identifying the transmitted symbols over fading channels with varying signal-to-noise ratios (SNRs). Traditional methods typically follow a two-step process: first estimating the channel using, e.g., the Minimum Mean Square Error (MMSE) estimator, and then performing symbol detection using the estimated channel. However, this approach can be computationally intensive and is impacted by the channel estimation quality. Data-driven approaches, such as deep learning models [1], [2] that directly learn channel estimators and symbol detectors, offer an alternative. However, deep neural networks (DNNs) require a large amount of data and often perform poorly in the low-data regime. More importantly, adapting pre-trained DNNs to new wireless conditions remains a challenge [3].

Advances in Transformer models, particularly decoder-only architectures like GPT [4], have demonstrated impressive performance across various fields. Recent result [5] shows that pre-trained Transformers can generalize to new tasks during inference through in-context learning (ICL), without requiring explicit model updates. Specifically, given an input of the form $(y_1, f(y_1), \ldots, y_n, f(y_n), y_{\text{query}})$, a pre-trained Transformer can approximate $f(y_{\text{query}})$ based on the provided context, where $(y_1, \ldots, y_n, y_{\text{query}})$ represents features and f can represent various classes of functions [6].

LF and CS were partially supported by the US National Science Foundation (NSF) under Grants AST-2132700, CNS-2002902, and ECCS-2029978. JY was partially supported by the US NSF under Grants CNS-1956276 and CNS-2114542.

Wireless symbol detection, which involves estimating transmitted symbols from noisy received signals, aligns well with the ICL frameworks. [7] introduces Transformers for this task using ICL, framing it as a regression problem with Mean Square Error (MSE) loss and achieving near-MMSE performance. Later works expand this framework: [8] extends it to MIMO systems, and [9] demonstrates its robustness in multi-user MIMO environments. Meanwhile, [10] employs language models to reformulate detection as a linguistic task. These advances highlight Transformers as a powerful tool for addressing wireless communication challenges.

Despite these successes, prior studies face limitations. Most approaches treat detection as a regression task, requiring MSE-based objectives and post-processing to map continuous outputs to discrete symbols. Additionally, many require ample pilot pairs, which may not be possible in practice, and large models increase inference costs, limiting real-world feasibility.

Inspired by decision feedback in wireless communication (e.g., decision feedback equalizers over multi-path fading channels), we propose a new prompt design by incorporating decision pairs. Specifically, we combine the current received signal with the model's detection as pairs, merging them with previous prompts to form a new, larger prompt for subsequent symbol detection. Our `DEFINED` model uses a carefully designed mixture training process to achieve high performance with limited pilots (sometimes only a single pilot) and maintain accuracy with sufficient pilots. Extensive experiments across modulation schemes validate our approach's effectiveness. To summarize, our main contributions include:

- We develop a Transformer model that jointly performs channel estimation and symbol detection. The key innovation in `DEFINED` is leveraging decision feedback with limited pilot data to expand the effective context length and enhance performance.
- We design a mixed training process for `DEFINED` that achieves high performance improvement with limited pilots and strong accuracy with sufficient pilot data, making the model adaptable to practical scenarios.
- We validate our approach with experiments across multiple modulation schemes.

II. System Model and Canonical Methods

A. Wireless Model

To more clearly illustrate our design, we consider a canonical receiver symbol detection problem over a standard narrowband wireless fading channel. Specifically, we focus on an $(N_r \times N_t)$ MIMO system, where the channel is represented by an $N_r \times N_t$ complex-valued matrix H_t at time t, following a distribution P_H. We normalize the channel coefficients such that each entry in H_t has a unit variance. The received signal at time t is expressed as: $y_t = H_t x_t + z_t$, where the channel noise $z_t \in \mathbb{C}^{N_r}$ is modeled as a complex additive white Gaussian noise vector with zero mean and covariance matrix $\sigma^2 I$. Each entry of the input vector $x_t \in \mathbb{C}^{N_t}$ is sampled uniformly at random from a given constellation set $\mathcal{X}$ (e.g., QPSK or 16QAM), and this modulation process is independently and identically distributed (i.i.d.) across both time and space. We normalize the signal to ensure a unit average total transmit power, i.e. $\mathbb{E}[\|x_t\|^2] = 1$. The average signal-to-noise ratio (SNR) at any receive antenna is given by $\text{SNR} = 1/\sigma^2$.

We focus on the *block-fading* channel model [11] in this paper, where the channel H_t remains constant over a coherent time period of T time slots, and is i.i.d. across different coherence periods. In other words, $H_t = h_l, \forall t = (l-1)T + 1, \cdots, lT$ for the l-th coherence period where h_l is drawn i.i.d. from P_H. Correspondingly, the data transmission is organized into frames, where each frame has a length that is at most T. The frame structure is designed such that the first k transmitted symbols are known and pre-determined *pilot symbols*, whose original purposes include assisting the receiver to perform channel estimation of h_l so that it can perform coherent symbol detection. In other words, based on the reception of a few pilot pairs $D_k = \{(y_1, x_1), \cdots, (y_k, x_k)\}$, the goal is to design a demodulator that accurately recovers the transmitted symbol $x_{k+1}, \cdots, x_T$ from the received signal $y_{k+1}, \cdots, y_T$ with high probability.

B. Canonical Methods for Symbol Detection

In the traditional approach, the receiver first estimates the channel using pilot signals, then performs symbol detection on the received signal y_t via hypothesis testing for each $t = k+1, \cdots, T$. Typically, the (Linear) MMSE estimator is used for channel estimation, and the MMSE channel estimate $\hat{H}$ is given by: $\hat{H}_k^{\text{MMSE}} = YX^\dagger(XX^\dagger + \sigma^2 I)^{-1}$, where X is the pilot matrix and Y is the received signal matrix. With the estimated channel, the transmitted symbol $\hat{x}_t$ is detected by projecting y_t onto the closest symbol in the modulation constellation $\mathcal{X}$ as: $\hat{x}_t = \arg\min_{x \in \mathcal{X}} \|\hat{H}x - y_t\|^2, \forall t = k + 1, \cdots, T$.

This two-step process treats channel estimation and symbol detection as separate tasks. Such decoupling can result in suboptimal detection, particularly under noisy conditions or limited pilot data [12]. Optimal estimators like MMSE rely on precise statistical models of the channel and noise, which are often hard to obtain. Additionally, these estimators are computationally intensive due to matrix inversions and posterior probability calculations, making them less appealing for real-time applications in high-dimensional systems.

To address these challenges, data-driven methods [13] have been explored for joint channel estimation and symbol detection. Deep learning architectures have shown promise [1], [2], [14]–[16], but their high data requirements [17] and poor adaptability to varying channel conditions without retraining limit their real-world applicability [3].

III. In-Context Learning-Based Symbol Detection

ICL for symbol detection leverages the structure of wireless communication frames, particularly in block-fading channels where channel conditions remain stable during the coherent time. Within each frame, pilot signals are followed by subsequent received signals, which naturally align with the Transformer architecture's strength in processing sequence-based inputs. The Transformer's ability to model dependencies among sequential data allows it to capture complex relationships within transmitted signals, making it highly effective for symbol detection tasks.

In this section, we introduce the ICL-based symbol detection method. We first formulate the symbol detection problem, and then present the ICL implementation using a popular Transformer model GPT-2, which also serves as the backbone of our proposed solution[1].

A. ICL-based Problem Formulation

Each ICL detection task τ corresponds to a latent channel H and a channel noise level σ^2, following the unknown joint distribution $P_\tau = P_H P_{\sigma^2}$. The ICL-based symbol detection does not have prior knowledge of the specific task τ and is provided only with a prompt $S_t^\tau = (D_k^\tau, y_t)$, consisting of k target pairs, $D_k^\tau = \{(y_1, x_1), \cdots, (y_k, x_k)\}$, which are sampled from the conditional distribution $P_{x,y|\tau}$ and serve as in-context examples for the current task τ, along with y_t, $\forall t = k+1, \cdots, T$.

As previously mentioned, each context pair (x_i, y_i) and the inference pair (y_t, x_t) are i.i.d. samples given task τ. For block-fading channels, the context set D_k^τ, also referred to as pilot signals in wireless communication, enhances the channel estimation, thereby improving the reliability and accuracy of data transmission. We note that a significant advantage of the proposed solution is that there is no need to change the existing frame structure or the design of pilot signals. Rather, the innovation is entirely at the receiver side where we leverage the pilot and decoded signals in a different way. This is an important advantage in practice as it allows for (backward) compatibility with the existing standard.

The goal of symbol detection is to identify the corresponding input signal x_t for the new query signal y_t from the same task. The ICL-based detection makes its decision as follows: $\hat{x}_t = f_\theta(S_t^\tau)$, where θ represents the parameters of the model. The detection for the query is measured by the Symbol Error Rate (SER), which is the frequency at which transmitted symbols are incorrectly decoded. The expected SER for the new query with k contexts, taking the expectation over the task distribution for $\forall k = 1, \cdots, T-1$, is defined as:

$$\text{SER}_k(\theta) = \mathbb{E}_\tau \mathbb{E}_{D_k^\tau, y_t|\tau} \left[f_\theta(D_k^\tau, y_t) \neq x_t \right]. \tag{1}$$

[1]We use GPT-2 with elaborated design choices as a concrete example throughout the paper. However, the proposed principle can be easily adapted to other Transformer architectures.

B. Vanilla In-Context Symbol Detection

Transformer models have emerged as powerful tools for symbol detection [18], leveraging their ability to capture long-range dependencies. This approach to wireless symbol detection was introduced in [7], [8]. The input-output structure is illustrated in Figure 1. With a causal masked self-attention mechanism, the model outputs the detection $\hat{x}_t$ at the corresponding position of y_t, relying only on known preceding contexts and the received signal. During the forward process, the Transformer solves $k+1$ detection problems for the same task τ, using an increasing number of pilot data points. Their results in [7], [8] demonstrate that the Transformer exhibits strong capabilities in symbol estimation within context, without requiring explicit model updates.

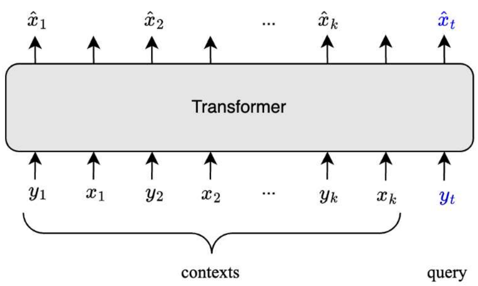


Fig. 1. Decoder-only Transformer architecture for ICL-based symbol detection with k pilots. Detection output applies to $\forall t = k+1, \cdots, T$.

IV. Decision Feedback In-Context Detection

The vanilla ICL approaches for symbol detection require sufficient context to achieve accurate estimation, which is often impractical in real-world scenarios. Pilot signals are costly and limited, reducing their adaptability for practical applications. For situations where the number of pilots is small, neither conventional two-stage (channel estimation then symbol detection) nor vanilla ICL solutions can achieve good performance. Furthermore, these approaches generally formulate the symbol detection task as a regression problem, as in [8], [9], [19], where a Transformer is trained to minimize the MSE loss. Although their models achieve performance comparable to the optimal MMSE estimator for x, an additional projection step is required to map the output to the appropriate transmitted symbol, leading to a mismatch and losing optimality.

In contrast, we directly define the problem as a classification task, enabling the model to jointly learn channel estimation and symbol detection while directly measuring the SER during inference. Additionally, we generalize the approach to effectively handle scenarios where pilot information is highly limited by sequentially feeding back the already decoded symbol pairs as *noisy pilots* and incorporating them as part of the prompt. Our model demonstrates robust performance even in challenging conditions with only a *single* pilot, outperforming previous ICL models that struggle with insufficient pilot data. At the same time, it maintains high accuracy when sufficient pilot data is available.

Inspired by the decision-feedback concepts in wireless communication, we propose the DEcision Feedback IN-ContExt Detection (`DEFINED`) method for symbol detection, as shown

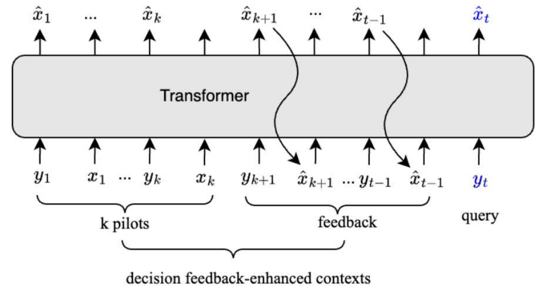


Fig. 2. `DEFINED` model architecture with k pilots and $(t-k-1)$ decision feedback contexts to detect x_t, $\forall t = k+1, \cdots, T$.

in Figure 2. The `DEFINED` model extends the prompt by incorporating the previously received signals and detection decisions alongside prior prompts to improve subsequent detections. While traditional decision-feedback equalizers (DFE) focus on inter-symbol interference (ISI), our study addresses narrowband channels without ISI. Nevertheless, decision feedback is effective here due to the latent common channel. Noisy feedback also provides valuable information, further refining model detection.

A. Model Parameters

Our specific Transformer model is designed with an embedding dimension of $d_e = 64$, $L = 8$ layers, and $h = 8$ attention heads, resulting in approximately **0.42 million** parameters, which is significantly smaller compared to large language models (LLMs) commonly applied in wireless communication tasks, such as those discussed by [10], [20]. For instance, even LLMs like GPT-J 6B contain over 6 billion parameters, making them approximately 14,000 times larger than our model. This compact size not only enables deployment on edge devices but also significantly shortens the inference time, enabling low-latency detection at the receiver.

B. Training Details

In this section, we describe our data generation process and the training of the `DEFINED` model. Training includes a pre-training phase to equip the model with general predictive abilities and speed up convergence, followed by fine-tuning to adapt the model to scenarios with limited pilot data.

1) Data Generation: We generate data according to the wireless communication model described in Section III-A. Specifically, we consider both SISO and 2x2 MIMO systems and explore various modulation schemes, including BPSK, QPSK, 16QAM, and 64QAM. For each wireless system and modulation task, we generate prompts consisting of sequences with T pilot pairs, with the maximum sequence length set to $T = 31$. Both systems operate under a Rayleigh fading channel, where the channel coefficient is sampled as $H \sim P_H = \mathcal{CN}(0, 1)$. The channel noise is sampled i.i.d. from a Gaussian distribution $z_t \sim \mathcal{CN}(0, \sigma^2 I)$. Within each block, the noise variance σ^2 is constant and shared across all noise samples. However, for different blocks, σ^2 is i.i.d. sampled from a uniform distribution P_{σ^2} over the range $[\sigma^2_{\min}, \sigma^2_{\max}]$ to improves adaptability. This variability applies only during training, while evaluation is conducted at fixed SNR levels for controlled performance assessment.

2) ICL Pre-training: We delve into the details of model training, which is divided into two phases, as shown in Figure 3. First, we define ICL-training and ICL-testing as operations on ground-truth data, represented by the clean prompt: $S_t^{\text{ICL}} = \{y_1, x_1, \ldots, y_{t-1}, x_{t-1}, y_t\}$, for $t = 1, 2, \cdots, T$. On the other hand, DF-training and DF-testing use iteratively decoded sequences with k pilot data and model decision feedback, which operate on the decision feedback prompt: $S_t^{\text{DF}} = \{y_1, x_1, \cdots, y_k, x_k, y_{k+1}, \hat{x}_{k+1}, \ldots, y_{t-1}, \hat{x}_{t-1}, y_t\}$, for $t = k+1, \ldots, T$, where each estimation $\hat{x}_t$ relies on the first k pilot points and prior model decisions.

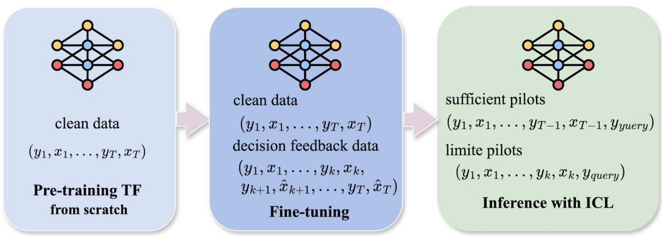


Fig. 3. The training process includes pre-training on clean data, followed by fine-tuning on a mixed dataset of clean and decision feedback noisy data. The model demonstrates strong performance, adapting to both limited and sufficient pilot scenarios during symbol detection (i.e., inference).

We define the loss functions for the ICL-training and DF-training models. The adopted loss function is the cross-entropy loss between the model's output and the ground-truth labels:

$$\mathcal{L}^{\text{ICL}}(\theta) = \frac{1}{NT} \sum_{i=1}^{N} \sum_{t=1}^{T} \text{loss}\left(f_\theta(S_{t,i}^{ICL}), x_{t,i}\right), \tag{2}$$

$$\mathcal{L}^{\text{DF}}(\theta) = \frac{1}{N(T-k)} \sum_{i=1}^{N} \sum_{t=k+1}^{T} \text{loss}\left(f_\theta(S_{t,i}^{DF}), x_{t,i}\right). \tag{3}$$

where θ represents the model parameters, and N is the number of samples.

DF-training is trained with limited pilot data and utilizes self-sampled labels, where noisy feedback is combined with previous contexts to generate new contexts. This process is time-intensive, as each detection and feedback step requires a model forward pass. Additionally, noisy data complicates convergence, which can cause the model to struggle to converge effectively. Training the Transformer with ICL-training and testing under DF-testing introduces a data mismatch: training uses clean data, while testing involves noise. Despite this mismatch, we observed that the Transformer's performance demonstrates significant robustness to noise during DF-testing and maintains acceptable performance. ICL-training is also about ten times faster than DF-training, as it eliminates data sampling and operates solely on clean data.

Considering all factors, our final proposed solution is to perform ICL-training first, followed by tailored DF-training. Here, ICL-training serves as pre-training, while tailored DF-training acts as fine-tuning, similar to the pre-training and fine-tuning process used in LLMs. Training epochs are carefully structured into two phases, as shown in Figure 3. In the first phase, the model converges just before reaching a plateau, at which point we transition to the tailored DF-training method. During this transition, a spike in the training loss is observed due to the shift in the training data distribution. As shown later, ICL pre-training not only accelerates convergence but also improves recognition of clean data and ICL-testing performance.

3) Decision Feedback Fine-tuning: Instead of vanilla DF training in the second phase, we employ a carefully designed fine-tuning process. The loss function is constructed as a linear combination of the previously defined losses in Equations (2) and (3), where α controls the balance between them, and the hyperparameter α is set to 0.7 during model training,

$$\mathcal{L}^{\text{fine-tuning}}(\theta) = \alpha \mathcal{L}^{DF}(\theta) + (1-\alpha)\mathcal{L}^{ICL}(\theta). \tag{4}$$

As explained in the pre-training phase, after ICL pre-training, the model is capable of general symbol detection, performing well on detections with clean data and, to some extent, on detections using the decision feedback method. Furthermore, in the fine-tuning process, the training loss is designed to emphasize decision feedback detection while retaining the model's ability to handle clean data.

Training on both clean and noisy data enhances the robustness of the Transformer model by exposing it to a more diverse dataset. Ultimately, we propose that **a single Transformer model** can be trained to perform both ICL-testing and DF-testing, making our `DEFINED` model adaptable for practical wireless systems. For example, in scenarios with sufficient pilot information, the model can operate in the ICL manner. However, in challenging situations – common in real-world applications – where pilots are limited and difficult to acquire, the model can utilize previous decisions to improve performance in subsequent symbol detection.

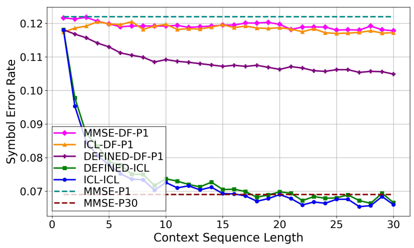
V. Experiment

We analyze `DEFINED`'s performance against baselines algorithms under both ICL-testing and DF-testing, represented by the "`DEFINED`-ICL" and "`DEFINED`-DF" curves, respectively. Our model not only performs well with sufficient pilot data but also exhibits significant improvements in limited-pilot scenarios by effectively utilizing noisy feedback. Moreover, `DEFINED` demonstrates strong performance in complex modulation tasks, highlighting the Transformer's ability to capture geometric structures within modulation constellations.

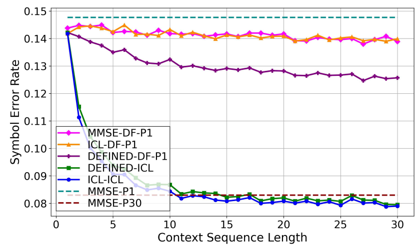
A. Baseline Algorithms

We introduce several baseline algorithms, including the ICL method, the MMSE algorithm, and MMSE-DF, a decision-feedback variant of MMSE.

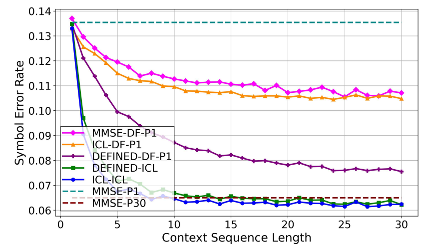
1) In-Context Learning: We train a Transformer from scratch using vanilla ICL-training and evaluate the SER for both ICL-testing and DF-testing, shown as "ICL-ICL" and "ICL-DF" curves in the figures, respectively.



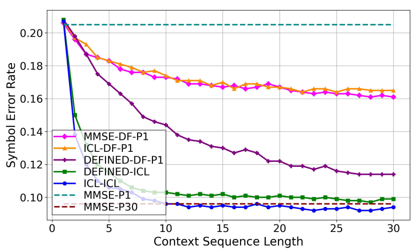
(a) SISO BPSK at SNR = 5dB with 1 Pilot, $\text{gain}_{\text{DF}} = 11.2\%$



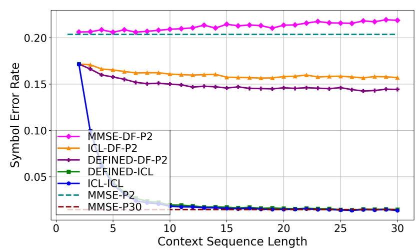
(b) SISO QPSK at SNR = 10dB with 1 Pilot, $\text{gain}_{\text{DF}} = 11.6\%$



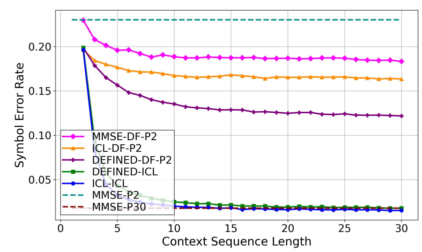
(c) SISO 16QAM at SNR = 20dB with 1 Pilot, $\text{gain}_{\text{DF}} = 43.7\%$



(d) SISO 64QAM at SNR = 25dB with 1 Pilot, $\text{gain}_{\text{DF}} = 45.6\%$



(e) MIMO BPSK at SNR = 10dB with 2 Pilots, $\text{gain}_{\text{DF}} = 16.8\%$



(f) MIMO QPSK at SNR = 15dB with 2 Pilots, $\text{gain}_{\text{DF}} = 38.6\%$

Fig. 4. SISO performance for BPSK, QPSK, 16QAM, and 64QAM with one pilot, and 2×2 MIMO for BPSK and QPSK with two pilots. The X-axis shows context sequence length, where the t-th point for $*$-ICL uses t ground truth pilots and $*$-DF-Pk uses k clean pilots and $(t-k)$ decision feedback noisy pairs.

2) MMSE Algorithm: This is a coherent detection algorithm, which first estimates the channel with the assistance of pilots and then performs detections for the subsequent received signals using the estimated channel. In the case of Rayleigh fading, both the channel and noise follow complex Gaussian distributions. The MMSE estimator for H is given by: $\hat{H}_k^{\text{MMSE}} = YX^\dagger(XX^\dagger + \sigma^2 I)^{-1}$. With the t-th received signal y_t, the transmitted symbol x_t is estimated by projection onto the closest symbol in $\mathcal{X}$:

$$\hat{x}_t = \arg\min_{x \in \mathcal{X}} \|\hat{H}_k^{\text{MMSE}} x - y_t\|^2, \forall t = k+1, \cdots, T. \quad (5)$$

With k pilot pairs, the mean SER is computed, shown as a horizontal line labeled ”MMSE-Pk”.

3) MMSE-DF Algorithm: MMSE-DF is an extension of the previous MMSE solution by sequentially using the decision feedback data as if they were new pilot pairs. Starting with k pilots, we compute the MMSE estimator of H and detect $\hat{x}_{k+1}$ using y_{k+1} as in Equation (5). The decision pair $(y_{k+1}, \hat{x}_{k+1})$ merges with the existing dataset, and iteratively used to detect each signal until $\hat{x}_T$. We plot the SER against the decision feedback-extended context sequence length.

B. Experimental Results

During testing, we sample 80,000 prompts to compute the mean SER. Results for BPSK, QPSK, 16QAM, and 64QAM in SISO, and BPSK and QPSK in 2×2 MIMO, plot SER against context sequence length, as shown in Figure 4. The t-th point represents the SER of $\hat{x}_{t+1}$ using t contexts, omitting the 0-th point (random guessing) due to high SER.

To quantify the SER improvement with increasing context length under DF-testing, we define the gain metric as: $\text{gain}_{\text{DF}} = \left(\frac{\text{SER}_k(\theta) - \text{SER}_{T-1}(\theta)}{\text{SER}_k(\theta)}\right) \times 100\%$, which represents the percentage reduction in SER as the context length increases from k to $(T-1)$, starting from k known pilots.

1) Comparison with Baseline Algorithms: The MMSE algorithm is shown alongside horizontal lines representing MMSE performance with k pilots and with full (30) pilots, respectively. The ICL-ICL line, for the model trained and tested with ICL method, shows that with 30 pilots, the Transformer slightly outperforms the MMSE algorithm during ICL testing. This improvement arises from model's ability to perform channel estimation and symbol detection jointly, leveraging the synergy between these tasks.

The DEFINED-DF line denotes the performance of our proposed model during DF-testing, showing a marked SER reduction as more decision feedback data is incorporated. For instance, in Figure 4d, for 64QAM with one pilot, the SER decreases from 0.206 to 0.112 as the feedback context sequence length increases, achieving a 45.6% SER gain. This confirms that the Transformer can effectively use noisy feedback to improve detections with limited pilot data.

Our model also performs well in ICL-testing, as indicated by the DEFINED-ICL line, which aligns closely with the ICL-ICL line. This result suggests that ICL pre-training, followed by carefully designed loss functions during decision feedback fine-tuning, allows the model to learn effectively from clean data. Additionally, our `DEFINED` model adapts well to real-world symbol detection, excelling with ample pilot data and performing effectively even with a single pilot.

The ICL-DF line, which represents a Transformer trained with ICL method but tested under DF conditions, performs significantly worse than our model, nearly coinciding with the MMSE-DF line. This observation highlights that models trained solely on clean data struggle with noisy feedback, as they simply treat all feedback data as clean. This underscores the importance of DF fine-tuning for handling noisy feedback.

2) Comparison with Different SNRs and Varying Pilot Lengths: At high SNR levels, reduced data noise enables more accurate detection from pilot data, enhancing `DEFINED` model performance and accentuating the downward SER trend. However, at very high SNRs, the already low initial SER limits further improvement. Our `DEFINED` model also performs robustly with minimal pilot data, including the extreme single-pilot cases. As pilot data increases, all algorithms show improved performance in DF inference.

3) Comparison with Different Modulation Schemes: We conduct experiments using BPSK, QPSK, 16QAM, and 64QAM modulations in the SISO system, representing classification tasks with 2, 4, 16, and 64 classes, respectively. As modulation complexity increases, the detection task becomes more challenging, but we observe greater performance improvements with complex schemes, as reflected by a more pronounced SER decrease with additional feedback data in our `DEFINED` model.

Analysis of the Transformer's output logit vector shows that nearly all of the incorrect detections occur within a small region around the ground-truth label in the constellation. This "typical error event" [11] suggests that even incorrect detections carry valuable information, *as the noisy label is often near the correct one, enhancing the model's detections*. Thus, as modulation complexity increases, the compact constellation set allows noisy feedback to provide more useful information, leading to better SER gains with added feedback data.

These findings demonstrate that our `DEFINED` model effectively captures the constellation set's geometry. It not only learns detections but also recognizes relationships between classes, often assigning higher probabilities to neighboring labels when errors occur. Due to inherent data noise – e.g., channel fading and additive noise – received signals with nearby latent labels in the constellation may overlap. As a result, the model sees adjacent labels as close neighbors, letting its detections retain valuable information, even when they are not entirely accurate.

4) MIMO Results: The results for MIMO systems with BPSK and QPSK using two pilots are shown in Figures 4e and 4f. Compared across different modulation schemes, SNR levels, our `DEFINED` model can still perform well for high dimensional detection problems. In fact, `DEFINED` exhibits a more pronounced decrease in SER with the inclusion of additional decision feedback data, leveraging the underlying communication system structure more effectively.

VI. Conclusion

Inspired by the decision feedback mechanism in wireless receiver designs, we proposed `DEFINED` to enhance symbol detection by incorporating decision pairs into the prompts of Transformer. Our approach achieved significant performance gains with limited pilot data while maintaining high accuracy with sufficient pilot data, demonstrating its adaptability for practical scenarios. Extensive experiments across various modulation schemes validated the robustness and flexibility of our model. These contributions highlighted the potential of Transformers, underscoring their capabilities for future wireless communication systems.

References

[1] C. Lu, W. Xu, H. Shen, J. Zhu, and K. Wang, "MIMO channel information feedback using deep recurrent network," *IEEE Commun. Letter*, vol. 23, no. 1, pp. 188–191, 2018.

[2] D. Neumann, T. Wiese, and W. Utschick, "Learning the MMSE channel estimator," *IEEE Trans. Signal Processing*, vol. 66, no. 11, pp. 2905–2917, 2018.

[3] O. Simeone, S. Park, and J. Kang, "From learning to meta-learning: Reduced training overhead and complexity for communication systems," in *2020 2nd 6G Wireless Summit (6G SUMMIT)*. IEEE, 2020, pp. 1–5.

[4] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[5] B. Mann *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901.

[6] S. Garg, D. Tsipras, P. S. Liang, and G. Valiant, "What can transformers learn in-context? a case study of simple function classes," *Advances in Neural Information Processing Systems*, vol. 35, 2022.

[7] V. Teja Kunde, V. Rajagopalan, C. Shekhara Kaushik Valmeekam, K. Narayanan, S. Shakkottai, D. Kalathil, and J.-F. Chamberland, "Transformers are provably optimal in-context estimators for wireless communications," *arXiv e-prints*, pp. arXiv–2311, 2023.

[8] M. Zecchin, K. Yu, and O. Simeone, "In-context learning for MIMO equalization using transformer-based sequence models," in *2024 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2024, pp. 1573–1578.

[9] ——, "Cell-free multi-user MIMO equalization via in-context learning," in *IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 2024, pp. 646–650.

[10] M. Abbas, K. Kar, and T. Chen, "Leveraging large language models for wireless symbol detection via in-context learning," *arXiv preprint arXiv:2409.00124*, 2024.

[11] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.

[12] C. Shen and M. P. Fitz, "MIMO-OFDM beamforming for improved channel estimation," *IEEE J. Select. Areas Commun.*, vol. 26, no. 6, pp. 948–959, August 2008.

[13] O. Simeone, "A very brief introduction to machine learning with applications to communication systems," *IEEE Trans. Cogn. Commun. Netw.*, vol. 4, no. 4, pp. 648–664, 2018.

[14] F. Liang, C. Shen, and F. Wu, "An iterative BP-CNN architecture for channel decoding," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 1, pp. 144–159, February 2018.

[15] F. A. Aoudia and J. Hoydis, "End-to-end learning for ofdm: From neural receivers to pilotless communication," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 1049–1063, 2021.

[16] S. Park, H. Jang, O. Simeone, and J. Kang, "Learning to demodulate from few pilots via offline and online meta-learning," *IEEE Trans. Signal Processing*, vol. 69, pp. 226–239, 2020.

[17] H. Kim, S. Kim, H. Lee, C. Jang, Y. Choi, and J. Choi, "Massive MIMO channel prediction: Kalman filtering vs. machine learning," *IEEE Trans. Commun.*, vol. 69, no. 1, pp. 518–528, 2020.

[18] W. Shen, R. Zhou, J. Yang, and C. Shen, "On the training convergence of transformers for in-context classification," *arXiv preprint arXiv:2410.11778*, 2024.

[19] A. Raventós, M. Paul, F. Chen, and S. Ganguli, "Pretraining task diversity and the emergence of non-bayesian in-context learning for regression," *Advances in Neural Information Processing Systems*, 2024.

[20] J. Shao, J. Tong, Q. Wu, W. Guo, Z. Li, Z. Lin, and J. Zhang, "WirelessLLM: Empowering large language models towards wireless intelligence," *arXiv preprint arXiv:2405.17053*, 2024.