

Towards a Meta-Learning Assisted Universal Neural Receiver: An Empirical Study

Yufan Wei[†], Wei Ye[†], Steven Sleder, Zhi-Li Zhang

University of Minnesota Twin Cities, Minneapolis, USA, 55455

{wei00329, ye000094, slede001}@umn.edu zhzhzhang@cs.umn.edu

Abstract—AI-native air interfaces, such as neural receivers, are pivotal for ensuring efficient communication in 6G networks. By leveraging the capabilities of neural networks, neural receivers aim to learn the characteristics of physical channels and configure numerous radio parameters more effectively than manual feature engineering. However, the practical deployment of these systems is hindered by high training costs and uncertainty surrounding their ability to generalize, which have not been sufficiently explored in the literature thus far. To fill this gap, we explicitly identify nine distinct data domains to investigate the generalization abilities of neural receivers. These domains arise from user behaviors, environmental scenarios, and base station configurations. We additionally explore the use of meta-learning techniques while simultaneously addressing the issue of data dimension mismatch through zero-padding to further bolster generalization. Comprehensive numerical experiments conducted using Nvidia’s Sionna physical layer simulator demonstrate that our neural receiver adapts to new scenarios with just a few seconds of fine-tuning, which yields competitive and sometimes superior performance compared to neural receivers trained directly on identical scenarios.

Index Terms—6G AI-native air interfaces, Physical layers, Meta-Learning.

I. INTRODUCTION

In the latest 5G specification Release 18, 3GPP initiates the study for the first time exploring the integration of AI/ML algorithms with air interfaces [1], [2]. This initiative aims to equip wireless devices with learning capability and better adaptability to dynamic environments. Unlike classical solutions, which primarily rely on manually pre-set rules or algorithms, AI-native air interfaces, such as neural receivers (NRXs), utilize neural networks to estimate real-time channel quality and autonomously adjust their signaling schemes [3]. Prototype experiments conducted by industrial researchers (including Nvidia, Nokia, Qualcomm, and Rohde & Schwarz) demonstrate the preliminary feasibility [4], [5] and suggest that their AI-native air interfaces could halve transmit power compared to 5G while providing the same bandwidth and data rates [6]. Undoubtedly, such advancement will greatly enhance the spectrum and energy efficiency of future 6G networks.

While NRXs have shown promising results in controlled laboratory environments, real-world wireless communication, especially for 5G and 6G networks, is notably more complex and heterogeneous: Mobile operators deploy diverse base station (BS) hardware, equipping a varied number of antennas and layouts, across different regions and may activate distinct

services at different times; The base station transmitters will not only dynamically decide varying coding rates, modulation orders, and MIMO layers [7], but also employ advanced functions to control power consumption, influencing the antenna activation [8]; Moreover, the contextual environment in which mobile users operate is also highly dynamic [9]. These factors and complexities are generally beyond the receiver’s control, posing challenges to the practical value of current NRXs, which requires training specific to the configuration of paired transmitter and usage scenarios.

Given this, we propose the concept of *universal NRX*, emphasizing its generalization capabilities. Specifically, in situations where resources are constrained and training from scratch is prohibitive, it should be able to accommodate different transmitter configurations, even those unseen during training, and rapidly adapt to varying contextual scenarios. This adaptability ensures the universal NRX achieves optimal performance across broad real-world conditions. Furthermore, its design prioritizes deployment across various user equipment and not just serve limited data types, catering to the wide range of 6G applications.

Unfortunately, to our knowledge, only limited in-depth exploration has been conducted in this area of the literature. None of the studies have discussed the impact of different data features, nor have they conducted comprehensive experiments to train and evaluate the NRX across the diverse data domains; refer to Sec. II-B. Such a gap leaves room for two intriguing research questions:

- 1) Which system features will significantly impact a neural receiver’s generalizability?
- 2) How to develop a universal neural receiver capable of rapid adaptation to various scenarios?

We explore these questions by identifying and investigating the impact of nine representative feature domains drawn from three categories, which collectively reflect the complexity and heterogeneity of the 5G and 6G networks: (1) Base station side factors, encompassing variables such as BS antenna numbers, antenna polarization, and adopted modulation order; (2) User equipment side considerations, which include user equipment capacity, user movement speed, and transmission data distribution; (3) Channel and scenario contexts, incorporating radio frequency, channel scenario types, and channel quality; listed in Table I. Our examination of these feature domains shows that their hindrance to NRX performance stems

[†] indicates co-primary author in this study.

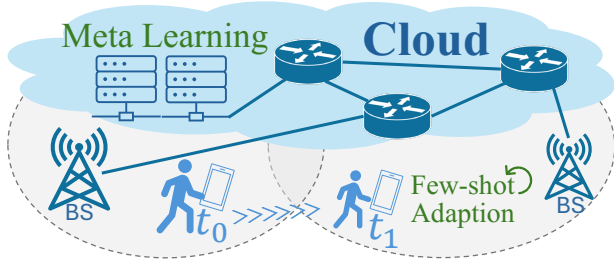


Fig. 1: A systemic view of meta-learning empowered neural receivers: with the pre-trained model, the user equipment can quickly adapt to fast-changing channels in real time.

from two primary issues: *dimension mismatch* and *distribution shift*. The former issue arises from the inherent restrictions of fixed neural network inputs, while the latter is due to the discrepancies between the training data source and more diverse scenarios encountered during testing. For details, we refer to Sec. III.

To address them and develop a universal NRX, we employ data augmentation through padding to address dimension mismatch and leverage Model-Agnostic Meta-Learning (MAML) [10] to mitigate distribution shift; see Sec. IV. We posit that the meta-parameter of an NRX can be trained on regional cloud servers and distributed to local users. Then, as users move between BS's service cell, UE can fine-tune parameters within a few seconds to adapt to the new scenario, as illustrated in Fig. 1.

To support this approach, we conduct comprehensive numerical experiments based on Nvidia's 6G physical layer simulator Sionna [11] to demonstrate the efficiency of our proposed universal NRX on both single-domain and multi-domain learning tasks. Our results indicate a bit error rate (BER) reduction of 96.28% on average by fine-tuning for less than a second when operating under the same channel quality as an ordinary NRX; see Sec. V for details.

Contributions. We summarize our key findings and contributions as follows:

- To the best of our knowledge, this is the first study that explicitly analyzes the generalizability of NRXs. We examine a total of three categories covering nine feature domains of communication systems perspective. We further identify two primary issues that prevent current NRXs from generalizing.
- We employ the zero-padding techniques to resolve the issue of dimension mismatch, and the MAML framework to address the data distribution shift problem, aiming to develop a *universal NRX*.
- Our comprehensive numerical experiment results demonstrate that the universal NRX consistently enhances performance, surpassing the ordinary one, especially on identified non-generalizable domains. The universal NRX can rapidly adapt to new scenarios within a few seconds of adaptation.

II. BACKGROUND AND STATE-OF-THE-ART

In this section, we provide a brief overview of wireless communication systems as well as state-of-the-art works in

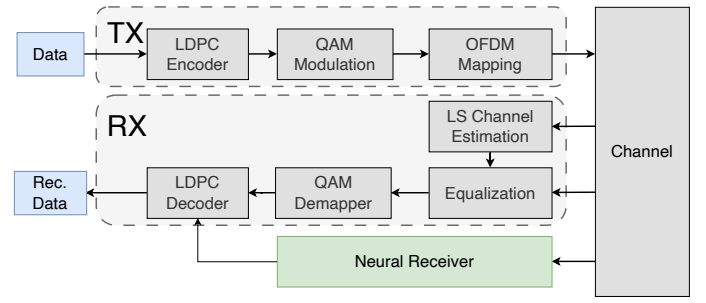


Fig. 2: An abstraction of wireless communication system and the role of neural receiver.

the neural receivers (NRXs).

A. Wireless Communication System Model

Fig. 2 abstracts a classical wireless communication system comprising a transmitter (TX) and receiver (RX). At the transmitter end, to ensure efficient transmission, the original bitstream undergoes a series of standardized processing steps such as encoding, modulation, and mapping. Specifically, the encoder adds redundant bits to the transmission block based on the encoding rate (a ratio of valid bits to total bits transmitted), the modulator encodes the digital data bits into the frequency domain according to the modulation order (an index indicates how many bits carried by per Hz), and the Multiple-Input Multiple-Output (MIMO) techniques, which leverages antenna's spatial multiplexing to transmit signal waves across multiple layers simultaneously. Higher coding rates, modulation orders, and transmission layers correspond to greater upper limits of transmission throughput, but also increase susceptibility to errors from the same channel noise [12], [13]. Thus, the transmitter's scheduler needs to make real-time decisions on the above parameters based on the channel quality feedback from the receiver.

At the receiver end, this process is reversed. The receiver will first estimate the physical channel H based on the reference signals. It attempts to estimate the channel $\hat{H} \approx H$ as in Eq. 1, using predefined signals x_{ref} and received signals y_{ref} , through the least squares (LS) problem with n being noise. Utilizing this estimated channel $\hat{H}$, the transmitted signal can be recovered, i.e., solving for x given $\hat{H}$ and y . The receiver then performs demodulation and decoding to recover the original digital data bits. In this study, we utilize an additive white Gaussian noise (AWGN) model where H is generated from the 3GPP Clustered Delay Line (CDL) channel model [14]. We refer the readers to [15]–[17] for more details about the system.

$$y = Hx + n \quad (1)$$

B. Neural Receivers

Neural receivers (NRX) aim to replace the above-mentioned classical receiver components, including channel estimation, equalizer, and QAM demapper, with a neural network, depicted by the green module in Fig. 2. The adoption of an NRX will not affect the way the current transmitter is employed.

TABLE I: Description of the feature domains studied in this paper. The marker ✓ indicates that NRX is generalizable to the setups (or scenario) with different parameters within this domain, see representative results in Fig. 3; while ✗ denotes NRX suffers poor generalizability toward this domain, see representative results in Fig. 4.

Domains	Descriptions	Generalizability
Moving speed	The movement speed (in m/s) of the receiver to account for the Doppler effect. We set {5, 10, 30} m/s representing "walking," "urban driving," and "highway driving".	✓
Data distribution	The distribution of data sources from which the data package samples. We consider {random, video, audio} with the latter two is downloaded from YouTube.	✓
Radio frequency	Frequency employed for transmissions between the transmitter and receiver. We consider {0.8, 2.6, 3.5, 28} GHz simulating the low-/mid-/high-band characteristics.	✓
Antenna polarization	The layout and polarization of the antenna adopted by the transmitter. We use {horizontal, vertical} layouts and {omnidirectional, directional} polarization.	✓
Modulation order	The number of information bits that are being carried per radio Hz. We consider {QPSK, 16QAM, 64QAM, 256QAM} denoting {2, 4, 6, 8} bits per Hz.	✗: shape mismatch.
CDL channel model	The channel model measured by 3GPP under different scenarios The model {A,B,C} is NLOS urban scenario, {D, E} is LoS rural and suburban.	✗: distribution shift.
Channel quality	The quality of the channel is measured by the normalized signal-to-noise ratio. We divide the range of [-3, 11] dB into intervals of 2 dB.	✗: distribution shift.
# of TX antenna	The number of antennas adopted by the transmitter used for the transmission. We set {2, 4, 6, 8} to accommodate various hardware limitations and power modes.	✗: shape mismatch and distribution shift.
# of RX antenna	The number of antennas adopted by the receiver used for the transmission. We set {1, 2, 3, 4} to accommodate various hardware limitations and power modes.	✗: shape mismatch and distribution shift.

The survey [3] summarizes emerging works that aim to improve the resistance of NRX to channel noise, thereby enhancing decoding efficiency. For example, [18] replaces the Fast Fourier Transform (FFT) processors in OFDM receivers with the complex-valued convolutional networks. While [19] introduces DeepRx, which employs convolutional neural networks to execute the entire receiver pipeline in a manner compatible with 5G standards. The NRX proposed in [20] addresses the distortions caused by the Doppler effect in extreme mobile scenarios, while [21] focuses on the multi-user scenarios. We note that these literature studies focus primarily on designing various neural network architectures to enhance performance towards ideal or specific scenarios but are missing the in-depth discussion toward the generalizability of the proposed NRXs as the applied scenarios vary.

Meanwhile, some works attempt to enhance the generalizability of NRXs but with a narrow focus: They either only consider a minimal set of synthesized channels [22]–[25] or utilize few-shot learning solely on neighboring or pilot symbols in the frequency domain [26]. These studies fail to adequately dissect the rich user context information and high heterogeneity present in real-world deployments, impeding NRXs from generalizing across scenarios. This gap presents an opportunity to (1) identify NRXs' generalizability over different data features; and (2) develop a *universal NRX* capable of rapid adaptation to various scenarios.

III. GENERALIZATION OF NEURAL RECEIVERS

To answer the first question, we identify and examine a total of nine feature domains in this section. We analyze each domain's effect on the generalizability of NRX under AWGN assumptions, outlining them in Table I. This analysis

is performed by varying the parameters of only a single domain per experiment, with all others remaining static. The simulation setup details is reported in Sec V-A.

A. Generalizable Domains

There are comprised of four domains in which a trained NRX remains performant. Specifically, we consider a range of moving speeds from 5 m/s to 30 m/s, radio frequencies spanning from low-band (0.8 GHz) to high-band (28 GHz), antenna layouts in both horizontal and vertical orientations, and polarization in either omnidirectional or directional modes. Additionally, we include input data distributions encompassing random 0-1 bits as well as real-world video and audio data.

We visualize the representative experimental results in Fig. 3, where a trained NRX is evaluated on setups using the other parameters in the same domain. we assess the generalizability of the NRX by examining the bit error rate (BER) as channel quality degrades as measured by the normalized signal-to-noise ratio E_b/N_o [27]. From the visualized results, we observe that the NRX achieves comparable performance across various E_b/N_o levels, indicating a degree of generalizability in those data domains.

B. Non-Generalizable Domains

We also identify five non-generalizable domains in which the NRX struggles, as reflected in Fig. 4. We hold that these issues stem from a combination of two primary sources: *data shape mismatch* and *distribution shift*.

Data shape mismatch is simply a result of the NRX's fixed input dimension, a constraint imposed by the architecture of its underlying neural network. Once initialized, its architecture cannot be easily altered, rendering NRXs trained on different parameters unadaptable. Due to this reason, as an example,

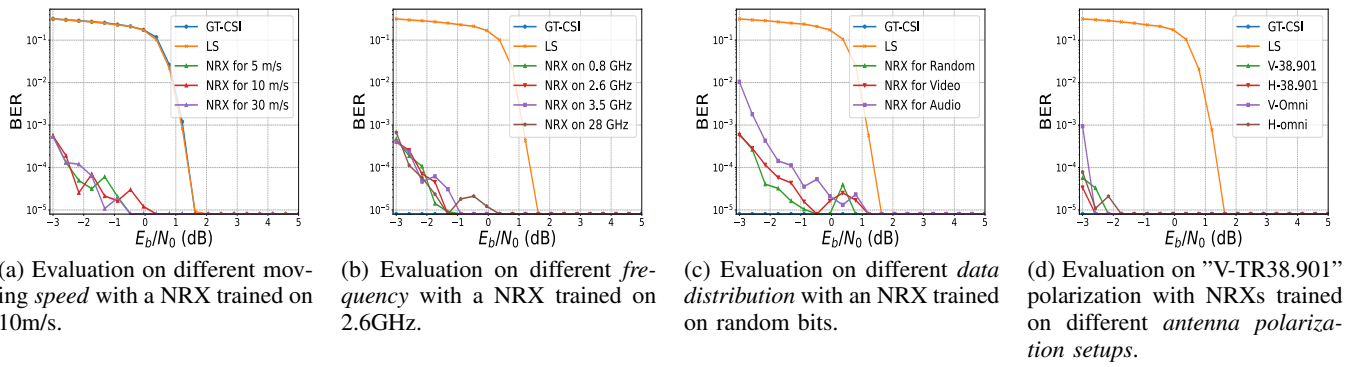


Fig. 3: Representative experimental results on *generalizable* domains, where a trained NRX remains performant. The baseline GT-CSI utilizes ground truth knowledge of the channel. The baseline LS-CSI uses classical least squares in channel estimation; see Sec V-A for illustration.

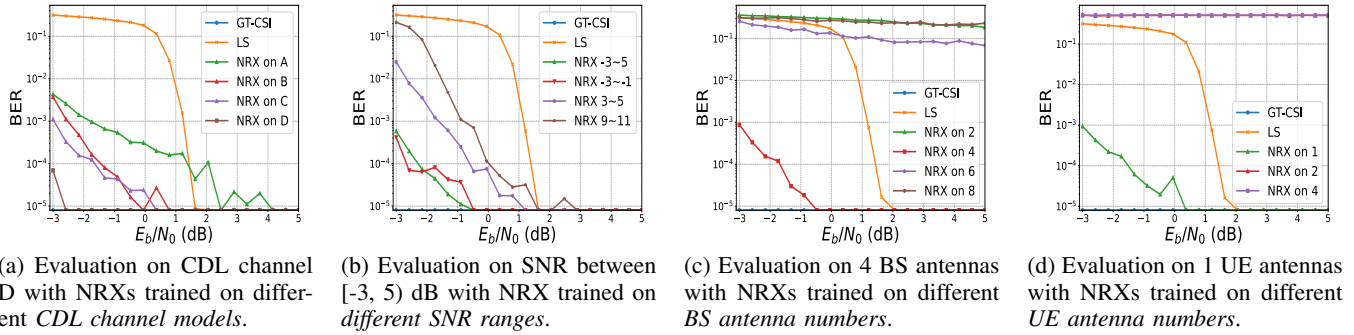


Fig. 4: Representative experimental results on *non-generalizable* domains, illustrating significant performance drops when domain parameter varies. Note that the experiment setup differs from Fig. 3.

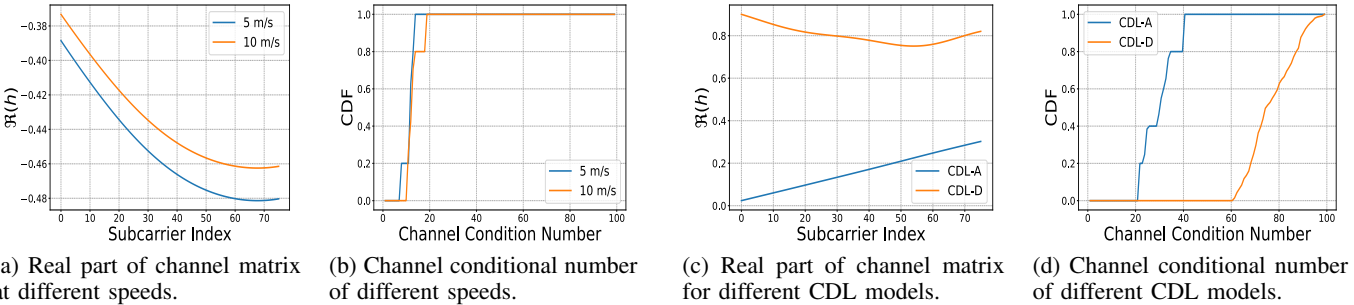


Fig. 5: Comparison of *generalizable* and *non-generalizable* feature domain from their channel characteristics.

an NRX trained on a system with 2 antennas (i.e., 2 streams) only cannot internally support a system with 4 antennas (i.e., 4 streams). It presents challenges in mobile networks, especially concerning heterogeneous BS deployments and compatibility with advanced functions, such as modulation, MIMO, power-saving, etc.

Additionally, significant distribution shifts that the NRX fails to capture are represented in various data domains, such as channel quality, CDL Channel Model, and the number of BS and UE antennas. We refer to Fig. 5 for a visual illustration, which compares the characteristics of the generated channel from two representative data domains. In the *generalizable* domain of speed, the real number part of each subcarrier¹

exhibits similarity across different speeds, yielding a consistent distribution of the channel condition number². Conversely, within the *non-generalizable* domain of the CDL model, variations in the real number part of each subcarrier across different CDL models result in disparate distributions of the channel condition number.

Such distinct channel characteristics lead to different levels of generalizability for an NRX among them, with representative results visualized in Fig. 4 where the significant performance drops are evident. For example, an NRX trained on the different CDL models exhibits unsatisfactory performance when handling the data that fits CDL model D (rural scenario

¹In OFDM system, each subcarrier is associated with an element in the channel matrix.

²Channel condition number quantifies the ratio of the maximum to the minimum singular value of the channel matrix, which assesses the quality of a communication channel.

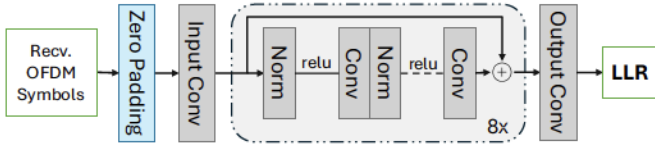


Fig. 6: Network architecture of neural receiver.

with line-of-sight). In some cases, the shift in data distribution causes NRX performance to be inferior even to the classical least-squares (LS) method.

It is worth noting that while Fig. 3 and Fig. 4 only visualize an NRX's ability to generalize across diverse setups within individual domains, combining multiple domains will significantly increase the complexity of the task. The combinatorial explosion will cause the scenario-specific NRX to perform poorly towards the highly heterogeneous network deployed in the real world.

IV. TOWARD UNIVERSAL NEURAL RECEIVER

To overcome the aforementioned challenges, we leverage zero-padding and model-agnostic meta-learning (MAML) techniques, which allow a trained model to be quickly adapted to new scenarios with dramatic data shifts, while needing only minimal tuning data. This enables a single, universal NRX to be deployed, bypassing the overhead of fully training and distributing independent domain-specific NRXs as they are needed. In the following, we formally introduce the problem task and present the details of our implementation.

A. Problem Statement

We denote the observed datasets as $D_{meta} = \{(D_t^{train}, D_t^{test})\}_{t=1:T}$ containing a total of T different learning tasks, each with specific training and test sets collected from the simulations by varying one or more domain parameters. Here, $D_t^{train} = \{(y_t^k, x_t^k)\}_{k=1:K_t}$ represents the training set of learning task t with a total of K_t samples, with D_t^{test} being the test set. For sample k in the learning task t , $x_t^k \in \mathbb{B}$ represents the original 0-1 data bit streams, while $y_t^k \in \mathbb{C}$ denotes the received signals with the dimension $[num_rx_ant, num_tx_ant, num_subcarrier, num_symbols]$. The learning task for the neural receiver f_θ is to accurately recover x_t^k from the received signals y_t^k , i.e.,

$$\arg \min_{\theta} \mathbb{L}(x^k, f_\theta(y^k)) \quad (2)$$

with $\mathbb{L}$ being the loss function of the bit error.

B. Universal Neural Receiver

As discussed in Sec. III-B, heterogeneous configurations and technologies employed by transmitters result in notable variation in the dimension of the receiver signal, y^k , across deployments. Meanwhile, the learning tasks span disparate domains (as outlined in Table I) with complex distributions, impacting the system behavior in non-trivial ways. Due to these issues, the model parameters θ of NRXs are specific to the task t of a single domain. To this end, our objective is

Algorithm 1 Training of Universal Neural Receiver.

Require: $p(\mathcal{T})$: distribution over tasks

Require: D_{meta} : dataset for training the meta-parameter

Require: α, β : step size hyper-parameters

- 1: Randomly initialize θ
- 2: **while** not done **do**
- 3: Sample batch of tasks $t's \sim p(\mathcal{T})$
- 4: **for** each task t **do**
- 5: Evaluate $\nabla_{\theta} l_t(f_\theta, D_t)$ with K examples.
- 6: Compute adapted parameters with gradient descent: $\phi_t = \theta - \alpha \nabla_{\theta} l_t(f_\theta, D_t^{train})$.
- 7: **end for**
- 8: Update $\theta \leftarrow \theta - \beta \nabla_{\theta} \sum_{t \sim p(\mathcal{T})} l_t(f_{\phi_t}, D_t^{test})$.
- 9: **end while**

to develop a universal neural receiver capable of (1) accommodating the varying dimensions of the received signals and (2) effectively mitigating the data distribution shifts.

Dimension agnosticism. Our neural architecture, as used in Sec. V, is shown in Fig. 6. Agnosticism to the input dimension is achieved by the module in blue, which performs zero-padding. This converts the data dimensions from $[batch_size, num_symbol, num_subcarrier, num_tx_ant]$ to $[batch_size, num_symbol, num_subcarrier, padded_length]$. The specific size of the padded dimension is left as a hyperparameter of the system. We apply padding on "num_tx_ant" since all of the shape mismatch problems we mentioned in Table I will eventually be reflected on that dimension.

The now uniform-length data vector is passed through a convolutional layer, eight residual blocks serving as hidden layers, and a final convolutional layer. Each convolutional layer utilizes a (3x3) filter with 128 channels over the OFDM resource grids, i.e., subcarriers and symbols, and batch normalization for stability. The final output of the NRX is the Log-Likelihood Ratio (LLR), sharing the same dimension as x_t^k , indicating the probability of each reconstructed bit being equal to 1.

Task domain agnosticism. We apply meta-learning to address distribution shifts, seeking the meta-parameter θ^* , which aims to achieve optimal performance in minimal fine-tuning steps and samples.

We outline the training process to find a performant meta-parameter θ^* in Alg. 1. Concretely, we first initialize the NRX with θ (Line 1). We then iterate through sample training tasks (Line 3). For each task t , we update its parameters using K samples, i.e., $(y_t^k, x_t^k) \sim D_t^{train}$, to determine the optimal parameters ϕ_t (Line 5-6).

$$\phi_t = \arg \min_{\theta} l_t(f_\theta, D_t^{train}) \quad (3)$$

with the loss function being binary cross entropy (BCE) between the original bits and the bits reconstructed by the neural receiver:

$$l_t(f_\theta, D_t^{train}) = \sum_k \text{BCE}(x_t^k, f_\theta(y_t^k)) \quad (4)$$

Once we obtain the parameters ϕ_t for each task, we evaluate their performance on D_t^{test} and update the meta-parameters (Line 8), i.e.,

$$\theta^* = \arg \min_{\theta} \sum_t l_t(f_{\phi_t}, D_t^{test}) \quad (5)$$

Using this approach, for any new task $(D_{t'}^{train}, D_{t'}^{test})$ we can rapidly obtain performant NRX parameters $\phi_{t'}$ which is initialized from the meta-parameter θ^* and fine-tuned on $D_{t'}^{train}$ using Eq. 3.

V. EVALUATION

In this section, we evaluate the proposed universal neural receiver, denoted as *MAML-NRX*. We outline the experimental setup and then demonstrate the meta-learning performance from both single-domain and multi-domain perspectives.

A. Experiment Setup

Methods. We consider four methods for comparison:

- *GT-CSI*: This method assumes access to ground truth knowledge of channel state information (CSI) or the channel matrix (i.e., H), which is then directly fed into the equalizer. The equalizer optimizes the linear minimum mean square error (LMMSE) between the received signal and the original signal by compensating for the distortion induced by the channel.
- *LS-CSI*: This method first estimates the channel based on the transmitted pilots using the least squares (LS) algorithm and then inputs this estimated channel into the LMMSE equalizer.
- *NRX*: The neural receiver, featuring data padding for dimension flexibility, as depicted in Fig. 6. It is directly trained on the D_t^{train} .
- *Joint-NRX*: It uses the same architecture as the NRX but adopts model-agnostic meta-learning training strategies (i.e., Alg. 1) on the D_{meta} without fine-tuning steps.
- *MAML-NRX*: It uses the same architecture and training strategy as the Joint-NRX but will conduct a limited number of fine-tuning steps when evaluated on an unseen scenario.

Hyperparameters. Unless otherwise specified, we use the following experimental hyperparameter configurations. The transmission bit stream is randomly generated from a Bernoulli distribution to train the neural receiver architecture depicted in Fig. 6. We employ a batch size of 64 and the Adam optimizer [28] with a learning rate of 1×10^{-3} . For the MAML-NRX, the training procedure uses the same learning rate for both inner-loop and outer-loop (i.e., $\alpha = \beta = 1 \times 10^{-3}$). To ensure convergence, we set sufficient epochs for all the experiments. For each epoch of MAML pre-training, we train for five steps to update the ϕ_t (i.e., line 6 in Alg. 1) and one more step to calculate the loss to update the meta parameter θ . For the unseen data in the evaluation stage, we fine-tune MAML-NRX with five epochs for adaption, which requires less than 1 second.

Evaluation. The evaluation spans different channel qualities, ranging from E_b/N_o of -3 dB to 5 dB with a step length of 0.4

dB. We conduct 100 Monte-Carlo simulations for each channel quality and with early stopping when the target number of transmission errors occur. We use the bit error rate (BER) defined as: $BER = \frac{\text{Error Bits}}{\text{Total Bits}}$.

Implementation. All experiments are implemented and conducted using NVIDIA's Sionna simulator [11], an open-source library for 6G physical layer simulation built upon TensorFlow [29]. The simulations are executed on a workstation with the AMD Ryzen Threadripper PRO 3995WX 64-Cores CPU and 3 NVIDIA RTX A6000 GPUs.

B. Single-Domain Meta Learning

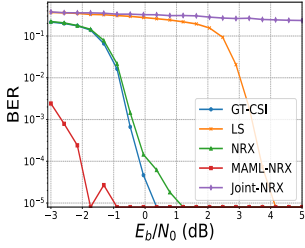
We first evaluate the performance of the MAML-NRX for single-domain scenarios with only 5 steps of fine-tuning over a few samples from the unseen dataset, costing 0.75 seconds. We compare it with a standard ordinary NRX and a Joint-NRX model as an ablation study. We visualize representative results within the domains of base station antenna number, user equipment antenna number, and channel models.

As shown in Fig. 7, MAML-NRX is pre-trained on D_{meta} comprising 4 and 8 antenna setups and fine-tuning on the 2 and 6 antenna systems. We compare it with both Joint-NRX, trained on D_{meta} , and NRX, which is directly trained on the 2 and 6 antenna systems. We see that MAML-NRX achieves competitive or even superior performance. Similar results are seen in Fig. 8 with MAML-NRX vastly exceeding the performance of the baselines. In this case, D_{meta} comprises 1 and 4 antenna setups. In Fig. 9 MAML-NRX continues to be superior for a D_{meta} comprised of CDL models A and E. Finally, Fig. 10 demonstrates that MAML-NRX remains competitive against the standard NRX when D_{meta} is scenarios 3 and 4 as defined in Table II.

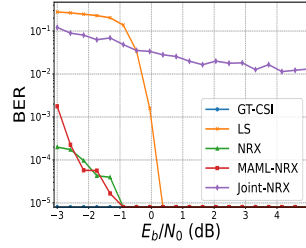
It is worth noting that for the unseen 2 antenna system, MAML-NRX reduces BER up to 99.99% (at $E_b/E_o = -1.7$ dB) with an average of 99.6% over the entire range of channel qualities. From another perspective, such improvement could lead to 2.9 dB of power savings to attain the same target bit error rate (BER) of 1×10^{-5} , meeting the 3GPP ultra-reliability requirements [30].

Similarly, as visualized in Fig. 8 and Fig. 9, MAML-NRX consistently achieves competitive or better performance than NRX which is directly trained on the same task setups. Specifically, when the receiver activates a varying number of antennas, MAML-NRX achieves a reduction in BER of up to 95.28% with an average improvement of 20.9% in scenarios where the receiver activates one antenna. In cases where the receiver activates two antenna setups, MAML-NRX achieves a reduction in BER of up to 99.99% with an average improvement of 99.96%. Across different channel models, MAML-NRX competes favorably, demonstrating significant improvements in both CDL channel models B and C, with maximum enhancements of 96.95% and 93.93%, respectively.

Furthermore, we observe that MAML-NRX not only surpasses the LS-CSI method but also, in certain instances, outperforms GT-CSI, which uses the ground truth knowledge of the channel matrix. We speculate that this could be attributed

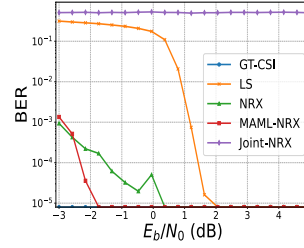


(a) Evaluation on Tx equipped with 2 antennas.

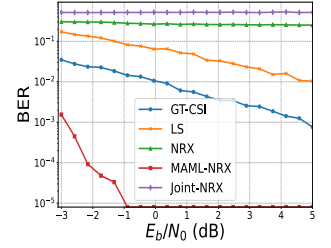


(b) Evaluation on Tx equipped with 6 antennas.

Fig. 7: MAML-NRX and Joint-NRX are pre-trained on the transmitter equipped with $\{4, 8\}$ antennas. NRX is trained on the same data domain as evaluation.

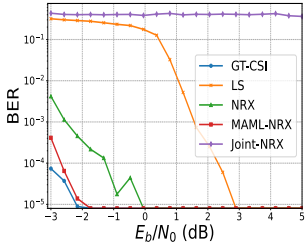


(a) Evaluation on Rx when only 1 antenna is activated.

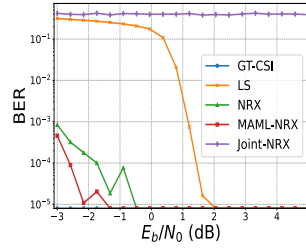


(b) Evaluation on Rx when 2 antennas are activated.

Fig. 8: MAML-NRX and Joint-NRX are pre-trained on the receiver activated $\{1, 4\}$ antennas. NRX is trained on the same data domain as evaluation.

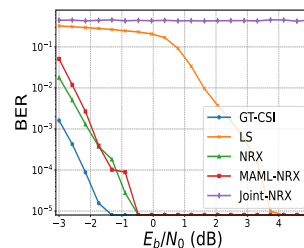


(a) Fine-tuning on B.

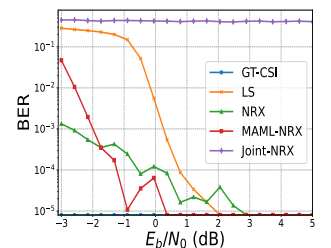


(b) Fine-tuning on C.

Fig. 9: MAML-NRX and Joint-NRX are pre-trained on the CDL channel model $\{A, E\}$. NRX is trained on the same data domain as evaluation.



(a) Fine-tuning on scenario 1.



(b) Fine-tuning on scenario 2.

Fig. 10: MAML-NRX and Joint-NRX are pre-trained on both scenarios $\{3, 4\}$. NRX is trained on the same data domain as evaluation.

TABLE II: Scenario with multiple domains.

Scenario	Combined Domains
1	3.5GHz; 5m/s; QPSK; CDL-A; bs_ant_num 6
2	2.6GHz; 10m/s; 64-QAM; CDL-B; bs_ant_num 4
3	0.8GHz; 20m/s; 256-QAM; CDL-C; bs_ant_num 2
4	28GHz; 5m/s; 521-QAM; CDL-D; bs_ant_num 8

to the neural network's superior learning capability exceeding the classical LMMSE equalizer. In summary, we conclude that adopting MAML can benefit NRX and significantly improve its performance by reducing the BER of 82.09% on average of different channel conditions across all the single domain learning tasks listed in Table I, which equals 1.7 dB gain in terms of power saving.

C. Multi-Domain Meta Learning

In addition to the single-domain meta learning, we extend to tasks spanning multiple domains. Table II presents representative scenarios involving the five domains used in our experiments, each containing different tasks, with selected results visualized in Fig. 10. We pre-train MAML-NRX on scenarios 3 and 4 to learn meta-parameters, a total of 10 different tasks, then fine-tune the learned meta-parameters on the evaluation scenarios 1 and 2, costing 0.72 seconds. In comparison, the NRX is directly trained on scenarios 1 and 2. From Fig. 10, we observe that MAML-NRX achieves up to 43.94% BER reduction in scenario 2, while it exhibits competitive performance compared to NRX in scenario 1. These results

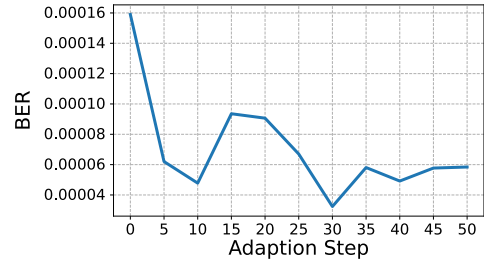


Fig. 11: MAML-NRX fine-tuning.

demonstrate that MAML-NRX effectively handles complex scenarios involving multiple-domain changes without requiring several days of training for a domain-dedicated NRX, providing nearly identical performance at best.

D. Efficiency of Fine-Tuning

To better understand the effect of fine-tuning, we utilize the experiment across different antenna numbers as an example. We visualize the change in bit error rate (BER) at $E_b/N_o = -2.6$ dB with varying adaptation steps, as depicted in Fig. 11. We observe that the pre-trained model of MAML-NRX demonstrates a level of generalizability even without any adaptation. In just 5 steps, it more than halves the bit error rate (BER), reducing it from 1.6×10^{-4} to 6×10^{-5} . We note that each adaptation step only utilizes 58k bits of data collected from the new scenarios and incurs an average cost of 0.14 seconds, which is feasible for mobile use cases.

VI. CONCLUSION

This paper aims to develop a universal neural receiver capable of rapidly adapting to new scenarios. We begin by examining the generalizability of current neural receivers across the representative data domains, including perspectives from base stations, user equipment, and channel contexts. Through extensive numerical experiments, we have identified two main challenges that lead to poor generalization performance: data dimension mismatch and distribution shift. To address these, we utilize zero-padding and MAML techniques. The evaluation on both single-domain and multi-domain tasks demonstrates that adopting MAML can significantly enhance the performance of neural receivers on unseen scenarios by reducing the bit error rate on average by 96.28%. It is competitive and often superior to that receiver exclusively trained for such scenarios.

Future works. Our analysis is based on the AWGN model, which may not accurately reflect real-world scenarios. We will consider more complex noise models and further set up the prototype software-defined radio platforms for future studies. Additionally, the acceleration of meta-model training will be a focus of future exploration. Lastly, we will also investigate correlations among domains and leverage more advanced learning techniques such as domain-disentangled meta-learning [31].

ACKNOWLEDGEMENT

We thank the anonymous reviewers for their suggestions and feedback. This research is supported in part by the National Science Foundation (NSF) under grants number 2128489, 2212318, 2220286, 2220292, and 2321531, as well as an InterDigital gift.

REFERENCES

- [1] "Study on artificial intelligence (ai)/machine learning (ml) for nr air interface," <https://portal.3gpp.org/desktopmodules/Specifications/>, accessed: 2024-05-14.
- [2] X. Lin, "An overview of ai in 3gpp's ran release 18: Enhancing next-generation connectivity?" *Global Communications*, vol. 2024, 2024.
- [3] N. Ye, S. Miao, J. Pan, Q. Ouyang, X. Li, and X. Hou, "Artificial intelligence for wireless physical-layer technologies (ai4phy): A comprehensive survey," *IEEE Transactions on Cognitive Communications and Networking*, 2024.
- [4] Dominique Loberg, "Enabling an ai-native air interface for 6g," <https://www.rohde-schwarz.com/enabling-an-ai-native-air-interface-for-6g/>, accessed: 2024-05-14.
- [5] "Towards an ai-native communications system design," <https://www.qualcomm.com/content/dam/qcomm-martech/dm-assets/documents/Towards-an-AI-native-communications-system-design.pdf>, accessed: 2024-05-14.
- [6] "6g technologies - ai-native air interface - nokia bell labs," <https://www.bell-labs.com/research-innovation/what-is-6g/6g-technologies/ai-native-air-interface>, accessed: 2024-05-14.
- [7] "Mcs / tbs / code rate in a nutshell," https://www.sharetechnote.com/html/5G/5G_MCS_TBS_CodeRate.html, 2019, accessed: 2024-05-14.
- [8] M. I. Rochman, D. Fernandez, N. Nunez, V. Sathya, A. S. Ibrahim, M. Ghosh, and W. Payne, "Impact of device thermal performance on 5g mmwave communication systems," in *2022 IEEE International Workshop Technical Committee on Communications Quality and Reliability (CQR)*. IEEE, 2022, pp. 1–6.
- [9] A. Narayanan, E. Ramadan, R. Mehta, X. Hu, Q. Liu, R. A. Fezeu, U. K. Dayalan, S. Verma, P. Ji, T. Li *et al.*, "Lumos5g: Mapping and predicting commercial mmwave 5g throughput," in *Proceedings of the ACM Internet Measurement Conference*, 2020, pp. 176–193.
- [10] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International conference on machine learning*. PMLR, 2017, pp. 1126–1135.
- [11] J. Hoydis, S. Cammerer, F. Ait Aoudia, A. Vem, N. Binder, G. Marcus, and A. Keller, "Sionna: An open-source library for next-generation physical layer research," *arXiv preprint*, Mar. 2022.
- [12] R. A. Fezeu, J. Carpenter, C. Fiandrino, E. Ramadan, W. Ye, J. Widmer, F. Qian, and Z.-L. Zhang, "Mid-band 5g: A measurement study in europe and us," *arXiv preprint arXiv:2310.11000*, 2023.
- [13] "5g nr throughput calculator," <https://5g-tools.com/5g-nr-throughput-calculator/>, Jun. 2019, accessed: 2024-05-14.
- [14] 3GPP, "3gpp ts 38.141-1 "base station (bs) conformance testing part 1: Conducted conformance testing", release 17," <https://www.etsi.org/deliver/etsi-ts/138100-138199/13814102/>, accessed: 2024-05-14.
- [15] A. Zaidi, F. Athley, J. Medbo, U. Gustavsson, G. Durisi, and X. Chen, *5G Physical Layer: principles, models and technology components*. Academic Press, 2018.
- [16] C. Johnson, *5G new radio in bullets*, 1st ed. Farnham (England): Chris Johnson, 2019.
- [17] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge, UK; New York: Cambridge University Press, 2005.
- [18] Z. Zhao, M. C. Vuran, F. Guo, and S. D. Scott, "Deep-waveform: A learned ofdm receiver based on deep complex-valued convolutional networks," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 8, pp. 2407–2420, 2021.
- [19] M. Honkala, D. Korpi, and J. M. Huttunen, "Deeprrx: Fully convolutional deep learning receiver," *IEEE Transactions on Wireless Communications*, vol. 20, no. 6, pp. 3925–3940, 2021.
- [20] J. Pihlajasalo, D. Korpi, M. Honkala, J. M. Huttunen, T. Riihonen, J. Talvitie, M. A. Uusitalo, and M. Valkama, "Deep learning based ofdm physical-layer receiver for extreme mobility," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2021, pp. 395–399.
- [21] S. Cammerer, F. A. Aoudia, J. Hoydis, A. Oeldemann, A. Roessler, T. Mayer, and A. Keller, "A neural receiver for 5g nr multi-user mimo," in *2023 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2023, pp. 329–334.
- [22] H. Mao, H. Lu, Y. Lu, and D. Zhu, "Roemnet: Robust meta learning based channel estimation in ofdm systems," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 2019, pp. 1–6.
- [23] S. Park, O. Simeone, and J. Kang, "Meta-learning to communicate: Fast end-to-end training for fading channels," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 5075–5079.
- [24] R. Li, O. Bohdal, R. Mishra, H. Kim, D. Li, N. Lane, and T. Hospedales, "A channel coding benchmark for meta-learning," *arXiv preprint arXiv:2107.07579*, 2021.
- [25] M. B. Fischer, S. Dörner, S. Cammerer, T. Shimizu, H. Lu, and S. Ten Brink, "Adaptive neural network-based ofdm receivers," in *2022 IEEE 23rd International Workshop on Signal Processing Advances in Wireless Communication (SPAWC)*. IEEE, 2022, pp. 1–5.
- [26] O. Wang, J. Gao, and G. Y. Li, "Learn to adapt to new environments from past experience and few pilot blocks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 9, no. 2, pp. 373–385, 2022.
- [27] Mar. 2024, page Version ID: 1213178766. [Online]. Available: <https://en.wikipedia.org/w/index.php?title=Eb/N0&oldid=1213178766>
- [28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [29] M. Abadi, A. Agarwal, P. Barham, E. Brevdo *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: <https://www.tensorflow.org/>
- [30] C.-P. Li, J. Jiang, W. Chen, T. Ji, and J. Smee, "5g ultra-reliable and low-latency systems design," in *2017 European Conference on Networks and Communications (EuCNC)*, 2017, pp. 1–5.
- [31] X. Zhang, Y. Li, Z. Zhang, and Z.-L. Zhang, "Domain disentangled meta-learning," in *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*. SIAM, 2023, pp. 541–549.