HORCRUX: Accurate Cross Band Channel Prediction

Avishek Banerjee*† banerjee.152@osu.edu The Ohio State University
Xingya Zhao* zhao.2053@osu.edu The Ohio State University
Vishnu Chhabra chhabra.67@osu.edu The Ohio State University
Kannan Srinivasan kannan@cse.ohio-state.edu The Ohio State University
Srinivasan Parthasarathy srini@cse.ohio-state.edu The Ohio State University

ABSTRACT

Recent advancement in Frequency Domain Duplexing (FDD) enables wireless systems to use different frequency bands for uplink and downlink communication without explicit channel feedback information. The current state-of-the-art approaches either estimate the underlying variables in the uplink channel or use an artificial neural network architecture to estimate the downlink channel from the uplink channel. However, such techniques fail to perform accurately in multipath-rich environments and environments unseen during training. This paper presents **HORCRUX**, a physics-based machine learning system that can be generalized and scaled to any environment while predicting downlink channels with high accuracy and applies to single-antenna and MIMO systems. Our approach uses multiple neural networks, trained on the standard wireless channel model, firstly to divide the uplink channel into smaller sub-channels and secondly to generate coarse estimates for the variables for each of the underlying sub-channels. Finally, we use an efficient and fast optimization framework to get fine-tuned variable estimates to predict the downlink channel. We implement our system using software-defined radios. Our evaluations show that HORCRUX performs ~8 dB better than state of the art in downlink channel prediction accuracy in diverse wireless environments. [1]

*Co-primary authors
†Current affiliation is Nokia Bell Labs. Work is done during previous affiliation.
[1]Code and data for HORCRUX are available at GitHub link

ACM MobiCom '24, September 30-October 4, 2024, Washington D.C., DC, USA

ACM ISBN 979-8-4007-0489-5/24/09…$15.00
https://doi.org/10.1145/3636534.3649343

CCS CONCEPTS

• **Networks** → **Wired access networks**; **Mobile networks**;
• **Computing methodologies** → **Neural networks**.

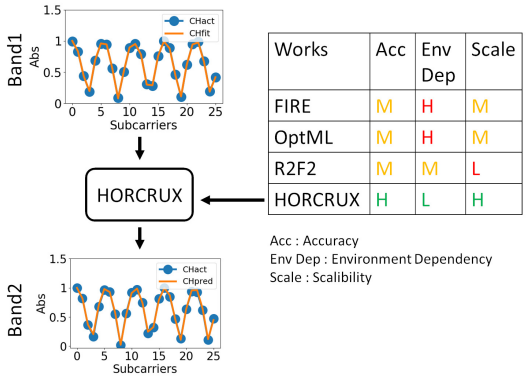
KEYWORDS

Physics-Guided Machine Learning, Wireless networks

1 INTRODUCTION

In recent years, advancements in wireless technologies have enabled high-throughput communication systems with the help of MIMO (Multiple Input Multiple Output). A key approach to increase the wireless system capacity is through the use of Multi-User MIMO (MU-MIMO) [34, 63] and Mega-MIMO [30, 33, 36]. These technologies allow multiple users to be served simultaneously by a single base station using multiple antennas. In applications like cellular networks, FDD [5] methods are used in conjunction with MIMO to simultaneously receive and transmit on different frequency bands. However, due to the lack of channel reciprocity across frequency bands, base stations require channel feedback from the clients to perform signal processing and modulation. This technique of channel feedback introduces expensive overhead and is unsustainable for MIMO systems [11, 12, 15, 22, 28, 35, 49, 58, 60].

State-of-the-art approaches such as R2F2 [54], OptML [11], and FIRE [35] addressed this challenge by removing the requirement of channel feedback and estimating the downlink channel based on the measured uplink channel. R2F2 and OptML are motivated by the fact that the underlying frequency-independent channel variables (multipath distance, attenuation, etc.) remain the same for both uplink and downlink channels. R2F2 proposes a novel signal processing technique to estimate the uplink's underlying parameters and thereby estimate the downlink channel. However, such a technique can be computationally expensive if the number of multipaths and antenna components is large. OptML [11], on the other hand, proposes a fast neural-network-based approach to estimate the channel variables and predict the downlink channel. FIRE [35] uses an end-to-end variational autoencoder (VAE) to estimate the downlink channel from



Works	Acc	Env Dep	Scale
FIRE	M	H	M
OptML	M	H	M
R2F2	M	M	L
HORCRUX	H	L	H

Acc : Accuracy
Env Dep : Environment Dependency
Scale : Scalibility

Figure 1: HORCRUX enables cross band channel prediction with high accuracy and scalability across environments, antenna sizes, and frequency bands. [H - High, L - Low, M - Medium]

the uplink channel. However, both OptML and FIRE estimate channels with low accuracy in multipath-rich environments if the training set differs from the testing environment. Such low accuracy in channel estimates hurts MIMO gains for multi-user settings.

To address this challenge, we introduce HORCRUX, as shown in Fig. 1, which achieves high channel estimation accuracy with zero feedback across any wireless environment and can be utilized across a single-antenna to a multi-antenna system. This work is motivated by the design framework of our previous work OptML [11].

MIMO systems take advantage of the multipath-rich environment to improve performance and capacity. Multipaths arriving at a separation of $\frac{c}{bw}$ (c is the speed of light and bw is bandwidth) can be resolved accurately. Previous efforts [11, 54] are inspired by this physics-guided intuition, and the resulting systems are trained based on understanding and resolving the underlying physical environment variables. For example, assume wireless channel of 20 MHz bandwidth 12 multipaths can be resolved within the delay spread of 1-200 meters [11]. In a particular environment, we measure the relationship between the observed wireless channels at different positions by a metric called path loss correlation. We used Pearson correlation [16] of the absolute values of the channel to calculate the path loss correlation. A realistic everyday environment should have uncorrelated path loss because of different combinations of multipaths (number of multipaths and their distances). Fig. 2 shows the relationship of path loss correlation with the underlying physical environment. Path loss of the channels observed is highly correlated (~0.7) when there is less variation in multipath and distance. However, with more environmental variation, the channels observed are uncorrelated (~0.2-0). It is to be noted that most channels in a realistic environment have uncorrelated path loss. In this work, we use the path loss correlation as a metric to estimate the diversity of wireless channels in experimented environments and clarify the performance gain of HORCRUX over the state-of-the-art systems.

State-of-the-art systems either use signal processing techniques (R2F2 [54]) or use physics-guided neural models to estimate these underlying multipaths (OptML [11]). However, our experiments show that such systems fail to perform accurately under uncorrelated path loss environments as they can only resolve up to 2-3 combinations of multipaths and do not perform well in unseen environments. A more recent effort FIRE [35] leverages a generative process with a VAE architecture for this purpose. However, it has been shown that such ideas do not work well if there is variability in the training data [61]. Specifically, our experiments show that if the wireless environment is complex, i.e., if the channel path losses are uncorrelated, FIRE fails to train accurately, resulting in erroneous estimations. Again, like OptML and R2F2, FIRE does not generalize to unseen environments. Fig. 3 offers a brief preview of the performance gain of HORCRUX against OptML, FIRE, and R2F2 under varying wireless environments. We observe that these contemporary systems perform poorly in environments with uncorrelated channels. However, HORCRUX continues to perform accurately under different wireless environments and outperforms state-of-the-art by ~8 dB gain in channel prediction accuracy when wireless channels are uncorrelated.

In this work, we propose a novel hierarchical approach (HORCRUX) that first subdivides the uplink channel into smaller sub-divisions based on underlying physical parameters of the channel and then utilizes a neural network model to estimate distances (motivated by OptML) on each subdivision. Finally, we use a fast and efficient optimization on each sub-division to estimate the downlink channel. A notable aspect of our framework is that the data we use to train our neural models is simulated (akin to a digital twin) and yet grounded by the underlying physical laws of the wireless channel. Unlike prior efforts, this implicit form of grounding [10] enables HORCRUX to generalize beyond the simulated environment it was trained on. We implement HORCRUX on WARP radios and compare its performance with FIRE [35], OptML [11], and R2F2 [54] in different indoor and outdoor test beds and simulations. Fig. 4 shows comparative results demonstrating HORCRUX outperforms the state-of-the-art.

Applications: Apart from widespread application in a multi-antenna base station for wireless systems, the zero feedback-based FDD system that HORCRUX subscribes to can have a significant impact in systems with a single antenna (smart home devices) or devices with antennas arranged in a random fashion (laptop or access points) that want to use different bands for uplink and downlink based on data-traffic and SNR (signal to noise ratio). Contemporary efforts, such as the FIRE architecture, will fail to support such variability in

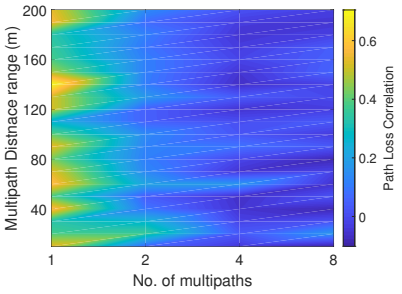


Figure 2: Path loss correlation in an environment decreases with an increase in the number of multipath components and their distances. Channel is highly uncorrelated when there are more than 2 multipaths. Everyday indoor and outdoor environments have uncorrelated channels.

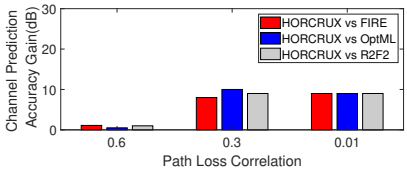


Figure 3: State-of-the-art predicts downlink channel with similar accuracy as HORCRUX in highly correlated path loss environments. However, in uncorrelated environments, HORCRUX enjoys ~8 dB gain compared to other systems. HORCRUX does not require any training in that particular environment.

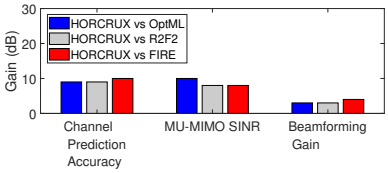


Figure 4: HORCRUX performs significantly better than the state-of-the-art in all the metrics. HORCRUX achieves ~8 dB gain in channel prediction accuracy and SINR in MU-MIMO and ~2dB improvement in beamforming gain compared to the state-of-the-art.

antenna orientation. HORCRUX, as it operates on a single antenna basis, can support all such applications.

Our paper makes the following contributions:

- We present a novel physics-guided hierarchical model to divide uplink channels into sub-channels based on multipath distances.
- HORCRUX utilizes a neural network model to estimate channel parameters in each sub-division. The data we use to train our neural models is simulated and grounded by the underlying physical laws of the wireless channel.
- To the best of our knowledge, we are the first to enable cross band channel prediction that can be generalized and scaled easily to any wireless environment without prior knowledge.

2 BACKGROUND

2.1 Channel Basics

In wireless communication, the receiver receives the composite signal from the transmitter after it has traveled across paths with different distances (d), attenuations (a), and reflections (ϕ). This composite effect of the environment that the signal undergoes is denoted as the wireless channel (h). Thus, for a signal transmitted at frequency f_i (wavelength λ_i), the multipath channel (h) can be represented as follows [11, 52]:

$$h_i = \sum_{n}^{N} a_n e^{\frac{-j2\pi d_n}{\lambda_i} + j\phi_n} \tag{1}$$

where N is the number of multipaths, and i is the subcarrier. Thus, for a single antenna, a multipath channel can be described by a set of 3-tuples, $\{(d, a, \phi)\}$ [11].

2.2 MIMO

MIMO systems exploit the multipath environment to simultaneously send and receive multiple data streams. For massive MIMO, the base station composes multiple antennas (K) which talk with multiple clients M ($M < K$) at the same time [37]. The data stream transmitted by the base station needs to be precoded to eliminate interference from other clients. Thus, the received signal can be expressed as

$$\mathbf{y} = \mathbf{H}^{\mathrm{T}}\mathbf{W}\mathbf{s} + \mathbf{n} \tag{2}$$

where $\mathbf{n}$ is noise, $\mathbf{s}$ is the $M \times 1$ vector of the transmitted signal, and $\mathbf{H}$ is the $K \times M$ channel matrix (each antenna of the base station to each client). Each client is assumed to have a single antenna for simplicity. $\mathbf{W}$ is the $K \times M$ precoding matrix. Similar to [35], in this work, we have used zero-forcing precoding technique [57]. Zero-forcing is a standard precoding technique where $\mathbf{W} = \mathbf{H}^{'}$, where $\mathbf{H}^{'}$ is the right pseudo-inverse of the channel matrix $\mathbf{H}$. Thus, it is essential to estimate the channel accurately. Otherwise, the clients will suffer from interference and can decrease the data rate and SINR (signal-to-interference-plus-noise ratio). In this work, we have used SINR as one of the metrics to evaluate HORCRUX on the MU-MIMO setup.

2.3 OptML Primer

OptML [11] is one of the state-of-the-art methods that enable a device having single or randomly arranged antenna elements to predict the channel to a client in frequency band F_1 based on the uplink channel observed from the client in frequency band F_2. To achieve that goal, authors develop a method for producing rough estimates of the distances of multiple paths by comparing the characteristics of the

detected channel with those created by signals that have traveled different distances.

$$P(d|h) = || \sum_{i}^{I} h_i e^{\frac{j2\pi d}{\lambda_i}} ||^2 \quad (3)$$

For a channel h measured over I sub-carriers, let h_i represent the channel at wavelength λ_i. Then, the likelihood of h containing a path from a distance d can be computed by Eq. 3 [11]. The distances with large values give the coarse estimate of the multipaths.

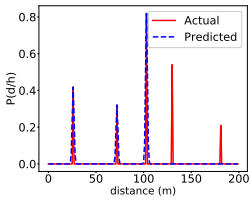


Figure 5: OptML fails to estimate all the multipath components. This results in erroneous channel estimation.

2.3.1 Neural Network Distance Estimator (NNDE). OptML uses a fully connected neural network to estimate the initial distance guesses. The input to the network is the uplink channel, and the output is a sparse vector containing the likelihood of the multipath components from different distances (ranging from 0 to 200 m) being a part of the channel. The neural net is trained on simulated data governed by the channel model as mentioned in Eq. 1. However, such a simplified network can, at most, estimate 2-3 multipaths using a single antenna and often underestimates the number of paths affecting the channel. Fig. 5 shows OptML cannot estimate all the components correctly.

2.3.2 Optimization Framework. OptML leverages an optimization framework to fine-tune the coarse estimates before predicting the cross band channel. They propose that the optimization can be framed in terms of $\{d_n\}$ (distance estimates) as the other variables can be estimated based on the least squares approach.

$$O(\{d_n\}_{n=1}^{N}) = ||H - DD^{'}H|| \quad (4)$$

where H represents the channel observed on a single antenna, D represents a matrix of size $I \times N$, $(D_{i,n} = e^{\frac{-j2\pi d_n}{\lambda_i}})$, where I is the number of subcarriers in the signal and N is the number of multipath components estimated by the NNDE. $D^{'}$ is the pseudo-inverse, and $D^{'}H = \overrightarrow{a}$ (the complex attenuation associated with each multipath component). Authors use a differential evolution algorithm and limit the search space within bounds ~20 m around the coarse estimates. However, the optimization becomes expensive with an increase in the number of multipath components. Using multiple antennas, OptML can resolve more paths but at the cost of computation time.

3 HORCRUX DESIGN

In this section, we propose our system HORCRUX. As shown in Fig. 6, HORCRUX takes the uplink channel as input and feeds into the Neural Network Channel Divider (NNCD) block. This block consists of parallel neural networks that divide the input channel into sub-channels based on the distances of the multipath components. The output from the NNCD blocks is then passed into the Distance Estimate Initiator block, which decides whether to initiate distance estimation for that predicted sub-channel. Following this, we introduce the mini-Neural Network Distance Estimator block (mNNDE), which contains parallel mNNDEs. These are similar to the NNDE introduced in OptML [11], however, simpler and light-weighted as trained on smaller datasets. Each of the mNNDEs focuses on a specific distance range and gives coarse estimates of distances of multipath components from that range. Then, HORCRUX uses the optimization framework similar to OptML to fine-tune distance estimates so that the channel based on the forecasted estimates closely fits the observed channel. However, HORCRUX converges faster as the bounds (Sec. 2.3.2) for each of the components are minimal (~4 m). Once converged, the estimates predict the cross band channel using Eq. 4. The neural nets are trained using simulated channel data on a single antenna system. For multi-antenna, HORCRUX will run simultaneously on each antenna.

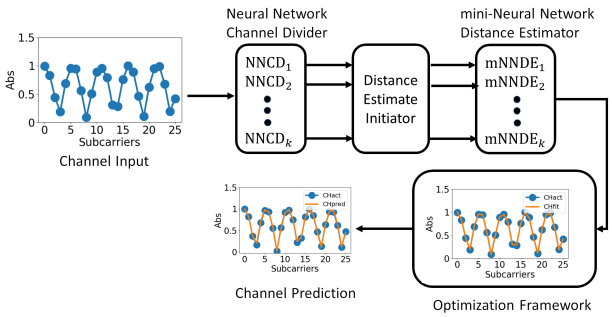


Figure 6: HORCRUX system overview

3.1 Neural Network Channel Divider

Instead of using the input channel to estimate coarse distances, we propose to divide the input channel into smaller subsections. This divide-and-conquer approach significantly improves the performance of HORCRUX as each divided channel has at most 1-2 multipath components (correlated

channels) and can therefore be easily estimated by the mNNDE. Say, for example, for a wireless channel has four multipath components, we can represent Eq. 1 as

$$h = \underbrace{a_1 e^{\frac{-j2\pi d_1}{\lambda}+j\phi_1}}_{\text{Zone 1}} + \overbrace{a_2 e^{\frac{-j2\pi d_2}{\lambda}+j\phi_2}}^{\text{Zone 2}} + \underbrace{a_3 e^{\frac{-j2\pi d_3}{\lambda}+j\phi_3}}_{\text{Zone 3}} + \overbrace{a_4 e^{\frac{-j2\pi d_4}{\lambda}+j\phi_4}}^{\text{Zone 4}} \quad (5)$$

As shown in Eq. 5, the purpose of the NNCD block is to divide the input channel h into separate zones. Each zone has a different neural network ($NNCD_i$) as shown in Fig. 6. The input to these NNCDs is the observed channel h, and the output is the channel (h_i) for that particular zone.

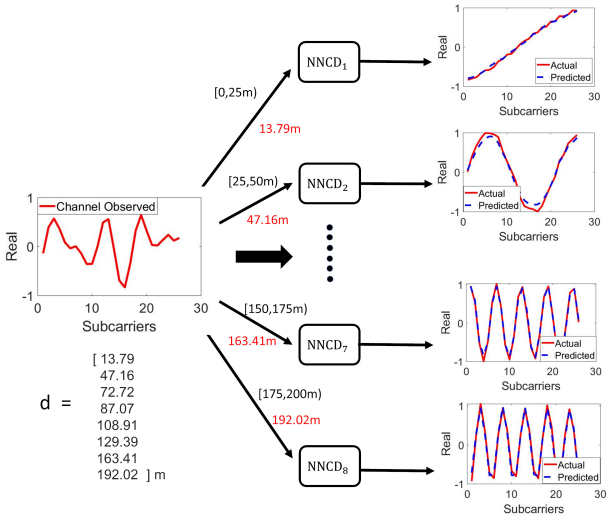


Figure 7: HORCRUX NNCD Block performance on a typically observed channel. NNCD can accurately divide the channel into corresponding zones. Only the real part of the channel is plotted for simplicity.

Physics-Guided Zone Division: The question is, how does one design the division of zones? To answer, we leverage the physics of the wireless environment.

i) The channels with multipath components of similar distances have a high probability of similar responses.

ii) The multipath components affecting the channel can be resolved based on the channel's bandwidth (c/bw).

Thus, for example, for a channel with 20 MHz bandwidth, we can resolve multipath components with distances (>15 m) apart. Therefore, for simplicity, we introduce eight zones for a 20 MHz channel and a maximum delay spread of 200 m. Each zone has a distance range of 25 m (>c/bw), i.e., $NNCD_1$ divides the channel into sub-channel formed by multipath components of distances $[0\ m, 25\ m)$, $NNCD_2$: $[25\ m, 50\ m)$ and similarly $NNCD_8$: $[175\ m, 200\ m)$. We assume that each zone has at most 1-2 multipath components. This assumption makes it easier to train each of the $NNCD_i$ blocks accurately, or it will try to fit noise without any multipath.

Fig. 7 shows the performance of the NNCD block for a typically observed channel. The observed channel consists of 8 multipath components (1 from each Zone), as shown as d in the figure. The $NNCD_i$ blocks take this observed channel as an input and divide the channel into the corresponding sub-division. As you can see in the figure, $NNCD_1$ predicts the sub-channel formed by the multipath component from Zone 1 (0-25 m), i.e., 13.79 m. Similarly, the rest of $NNCD_i$ blocks perform their channel division. In Fig. 7, for simplicity, we have shown the prediction of $NNCD_1$, $NNCD_2$, $NNCD_7$, $NNCD_8$ only. As you can see, our channel prediction is near identical to the actual subdivided channel.

As mentioned, the NNCD blocks are separately trained on simulated channel data. Fig. 8 shows the prediction accuracy of each of the $NNCD_i$ blocks. Similar to [35], we define channel prediction accuracy as

$$Accuracy\ (dB) = -10log_{10}\left(\frac{||H - H_{gt}||^2}{||H_{gt}||^2}\right) \quad (6)$$

where H is the predicted sub-channel and H_{gt} is the ground truth sub-channel. As seen in Fig. 8, all the NNCD blocks predict the sub-divided channel with a mean accuracy > 15 dB. However, as you can see the edge blocks $NNCD_1$ and $NNCD_8$ perform better (> 20 dB). The performance degrades as we move towards the middle blocks due to error propagation in dividing the channels. Adjacent zones since having multipath components with similar distances (< 25 m) can be erroneous in estimation. Edge blocks (e.g., $NNCD_1$) have error propagating from only one adjacent block $NNCD_2$, thus can be trained accurately compared to middle blocks (e.g., $NNCD_3$) where error propagates from both adjacent zones. However, this error propagation is nominal (< 3 dB) and does not hurt downlink channel prediction accuracy.

3.2 Distance Estimate Initiator

One of the assumptions made in training for the NNCD is that there is always at least one multipath from each section. This assumption helps to train the neural network models without fitting noise in case there is no multipath in a particular section. However, in actual experiments, such an assumption is not guaranteed to hold. Thus, even in the absence of a multipath component from a zone, the NNCD of that zone will still try to divide the channel into a corresponding sub-channel. If this sub-channel is further used for distance estimation, it will generate false distance estimates. Thus, if not taken care of, it will result in erroneous channel prediction in the downlink band. To overcome this, we introduce the Distance Estimate Initiator block. A key

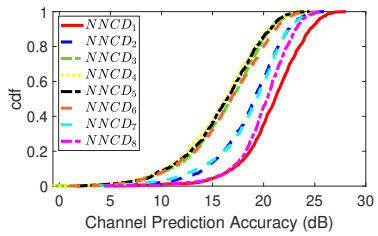


Figure 8: HORCRUX NNCD Block performance on a typically observed channel. NNCD can accurately divide the channel into corresponding zones.

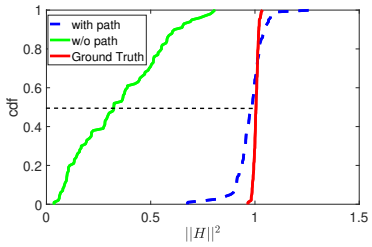


Figure 9: NNCD estimates sub-channel with low power in the absence of multipath components. This observation is used to estimate α.

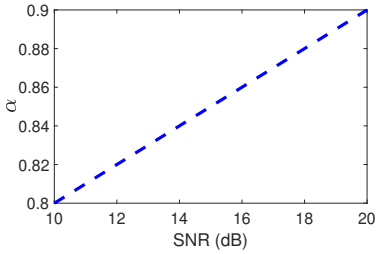


Figure 10: Distance Estimate Initiator decides to initiate $mNNDE_i$ if $||H_i||^2 \geq \alpha$. α varies with SNR linearly.

observation here is that the corresponding NNCD predicts the sub-channel with a very low mean absolute power (noise) if there is no multipath component from a particular zone. Fig. 9 shows the cumulative distribution function (CDF) plot for a $NNCD_i$ with and without a path. One can observe that the mean absolute power of the predicted sub-channel in the presence of a component is around 1 mW (mean CDF), similar to the ground truth sub-channel. However, in the absence of a multipath component, the predicted sub-channel of the $NNCD_i$ is ~0.4 mW. Thus, we define a threshold α that decides whether to initiate the distance estimation for that particular sub-channel.

$$f_i = \begin{cases} 1, & \text{if } ||H_i||^2 \geq \alpha \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where f_i decides whether to initiate the $mNNDE_i$ (Sec. 3.3). Note that α will change with different signal SNR. Fig. 10 shows the α values for different SNRs. Also note that the α is chosen as the minimum of all the $NNCD_i$ blocks, as the middle blocks can suffer a bit from erroneous estimations as mentioned in Sec. 3.1.

3.3 mini-Neural Network Distance Estimator

Each of the $NNCD_i$ blocks is accompanied by a mini-Neural Network Distance Estimator ($mNNDE_i$) as shown in Fig. 6. HORCRUX leverages a fully connected neural network like that introduced in OptML [11]. However, the model size is much smaller than OptML since each $mNNDE$ only estimates 1-2 multipath components, focuses on a smaller distance range of 25 m, and can be trained on a smaller dataset. These $mNNDE_i$ blocks are separately and independently trained on physically governed simulated data based on Eq. 1 and are not dependent on the training of $NNCD_i$.

Fig. 11 shows the performance of mNNDE blocks. We used the example discussed in Fig. 7. After being subdivided by the $NNCD_i$ blocks, the observed input channel is fed into $mNNDE_i$. f_i in Fig. 11 shows the Distance Estimator Initiator for each flow. Each $mNNDE_i$ takes the predicted sub-channel of $NNCD_i$ as input and gives the coarse estimates of the distance of the multipath components. Fig. 11 shows our mNNDE blocks accurately estimate the multipath components. Thus, using this technique, we can resolve 8-10 components for a 20 MHz channel, while OptML can only support two to three multipath estimations using a single antenna.

3.4 Optimization Framework

This section describes how HORCRUX fine-tunes the coarse estimates of the distance estimator blocks before predicting the channel in different frequencies. We use the same formulation introduced by OptML, mentioned in Eq. 4. A differential evolution algorithm is used for the optimization. However, because of our design approach, we can constrict the bounds for each coarse distance estimate (d_n) to about 4 m, ~4x faster than OptML. In other words, HORCRUX can converge more quickly even if the number of multipath components is eight. Fig. 12 shows the result for the optimization framework. As we can see, HORCRUX fits the observed channel accurately by estimating the channel variables $\{a_n, d_n\}$. These variables are used to estimate the cross band channel. HORCRUX predicts the downlink channel with high accuracy as shown in Fig. 12.

3.4.1 Channel Prediction Algorithm. Algorithm 1 shows the overall algorithm for HORCRUX channel prediction. It takes N as the number of zones as input and the corresponding $NNCD$ and $mNNDE$ blocks. It runs for all antenna elements (K) in parallel. For each antenna (j), $NNCD_i$ blocks run parallel to subdivide the uplink channel (H^j_{UL}) into H_i. Distance Estimate Initiator calculates f_i. If true, each of the $mNNDE_i$ blocks gives the coarse distance estimates. $\{d_N\} = \{d_1, d_2 \dots\}$ estimated by the $NNDE_i$. The initial estimates are then replaced with bounded variables $\{d\}_{bounded} = \{d1 \pm b, d2 \pm b \dots\}$ (getBounded in Algo. 1) to reduce the

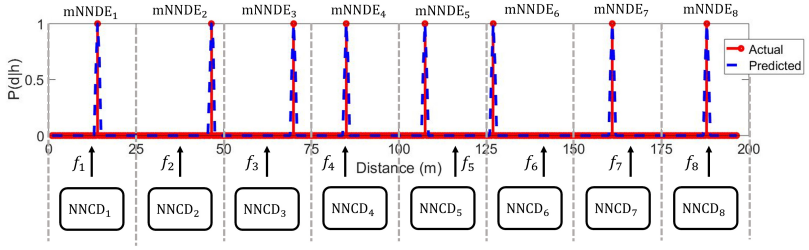


Figure 11: HORCRUX mNNDE blocks provide accurate distance estimates. This unique design allows HORCRUX to estimate multipath components accurately. Each $mNNDE_i$ takes the output of $NNCD_i$ as input.

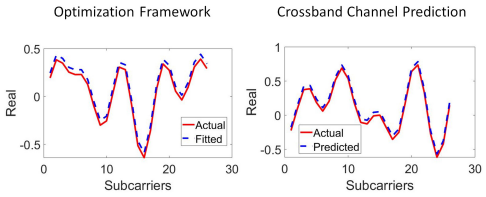


Figure 12: HORCRUX fits the observed channel using our optimization framework and then, based on the estimates, predicts the cross band channel accurately.

search space and are then used in the optimization algorithm. The fine tuned estimates $\{d\}_{final}$ are then used to compute $\overrightarrow{a}$. Finally, HORCRUX computes the downlink channel H^j_{DL}.

Algorithm 1: Channel Prediction Algorithm

Input: N, $\{NNCD_{1-N}\}$, $\{mNNDE_{1-N}\}$, λ_{UL}, λ_{DL}, H_{UL}, b, α
Output: H_{DL}
for all $j \in K$ **do in parallel**
  for all $i \in N$ **do in parallel**
    $\{H_i\} = NNCD_i(H^j_{UL})$
    Calculate f_i based on Eq. 7
    if f_i ==1 **then**
      $\{d_i\} = mNNDE_i(H_i)$
    end if
  end for
  $\{d\}_{bounded}$= getBounded ({ d_N},b)
  $\{d\}_{final}, error$ = Optimize ($\{d\}_{bounded}, H^j_{UL}$)
  D_{UL} = getMatrix ($\{d\}_{final}, \lambda_{UL}$)
  $\overrightarrow{a}$ =$D'_{UL} H^j_{UL}$
  D_{DL} = getMatrix ($\{d\}_{final}, \lambda_{DL}$)
  $H^j_{DL} = D_{DL}\overrightarrow{a}$
end for

3.5 Training data

Each of the $NNCD_i$ and $mNNDE_i$ is trained separately on physically grounded simulated data (Eq. 1 is used to generate the wireless channels). The parameters for generating data for each block are similar to OptML [11].

- RF parameters: center frequency and bandwidth
- Component distances $d \in [dmin, dmax]$
- Attenuation, $a \in (0, 1]$
- Phase, $\phi \in [-\pi, \pi]$

The center frequency is chosen based on the uplink of the Wi-Fi channel, and the bandwidth is 20 MHz. Each multipath is represented by a set of 3-tuples, $\{(d, a, \phi)\}$. d varies between 0 to 200 m. Note that the wireless channels generated are highly diverse and uncorrelated with a path loss correlation value ~0, which resembles a dynamic wireless environment.

NNCD: We separately simulate a sub-channel for each zone based on the above variables. The sub-channel is simulated based on the $\{(d, a, \phi)\}$ of the multipaths of that particular zone. Each zone has at least one multipath component. The input channel to the NNCD is calculated by summation (normalized) of all the sub-channels for that particular data point. The output is the generated sub-channels. Fig. 7 shows the input and output of the NNCD blocks.

mNNDE: The simulated sub-channels for NNCD are used as input to mNNDE. The output is the sparse vector of true distances for that sub-channel. The number of zones for HORCRUX depends on the channel's bandwidth. Throughout this paper, we used 20 MHz bandwidth and eight zones.

Digital Twin Advantage: A substantial benefit of our approach is leveraging a physically grounded digital twin environment to train our neural models. Such ideas have been recently explored in various contexts such as manufacturing [6, 69], intelligent transportation [25], and aeronautical designs [38]. This simulated environment, guided by accurate real-world physical models, has significant advantages over training on data within a specific environment [35]. First, unlike models trained on specific environmental conditions that required specialized detection and resolution strategies for out-of-distribution observations, our digital twin-based training framework can sidestep this issue, allowing it to generalize to arbitrary environments. Second, since these signals are being generated with ground truth labels, we do not require manual ground truth labeling by experts, which can be both expensive and prone to error. Third, the process facilitates an implicit form of grounding for our models [10]. Finally, the experimental results (see Sec. 5) demonstrate that such a strategy can be enhanced and cross-validated with real-world information.

4 IMPLEMENTATION

To evaluate the performance of HORCRUX in the real world, we build a prototype with WARP v3 software-defined radio

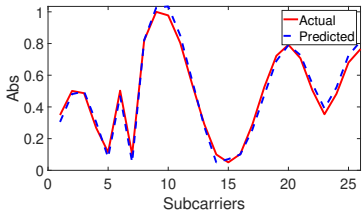


Figure 13: HORCRUX predicts the absolute value of the channel accurately across subcarriers.

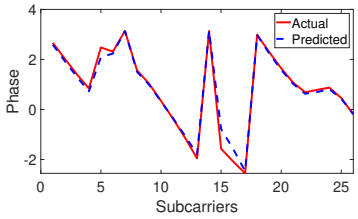


Figure 14: HORCRUX predicts the phase value of the downlink channel accurately across subcarriers.

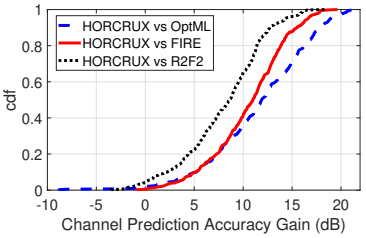


Figure 15: HORCRUX outperforms baselines in LOS conditions for a 2-antenna base station in a small room environment.

platform [1]. We tested our system on a 2-antenna and a 4-antenna base station, talking with two clients. Experiments were conducted in different wireless environments (a sizeable indoor office space, smaller indoor household) in i) LOS (line-of-sight) ii) NLOS (non-line-of-sight) and iii) moving client conditions. The wireless environments had varied path loss correlations. The details of the path loss correlation of each environment are shown in Fig. 29. We used path loss correlation as a metric to show the diversity of channels in the wireless environments. Clients talk with the base station using Wi-Fi 802.11n protocol. The base station uses the Long Training Symbols (LTS) to do channel estimation after Carrier Frequency Offset (CFO) and Sampling Frequency Offset (SFO) correction. The base stations have the antennas separated by around half wavelength (~6 cm) for the 2.4 GHz band. 20 MHz bandwidth and 64 OFDM subcarriers were used. Like FIRE [35], we split the channel measurements into two halves (26 subcarriers after removing guard bands). We used the first half as uplink and estimated the other half using HORCRUX.

FDD: We also evaluated our system in the FDD hardware setup using WARP nodes, where the uplink and downlink happen on different frequencies and across other devices.

NNCD achitechture We implemented the NNCD models within Keras [19]. Each NNCD comprises two hidden layers, each with 128 neurons with an Exponential Linear Unit (ELU) activation function. The models are trained separately on 100K data points with the same input channel and target sub-channels. The input and output to the neural net is a linear array of $[2 \times N_{fft}]$ (real and imaginary part of the complex channel with N_{fft} subcarriers). We generated various training channels with different SNRs, and each NNCD zone had at least one multipath component.

mNNDE architecture: Similar to NNCD, each of the mNNDE models is trained separately on 100K data points, each of which represents input channels generated by multipath components of that particular zone and target vector of distance estimates (range of 25 m). Each of the models is composed of 5 hidden layers, with 200 neurons on each layer.

Optimization: For optimization, we use Python [11, 29, 53]. Based on path resolution, we limit the maximum multipath component to 12 for 20 MHz. The trained models ($8 \times$ NNCD and $8 \times$ mNNDE) take less than 7 MB of disk space (similar to OptML).

5 RESULTS

In this section, we present an empirical evaluation of HORCRUX. We compared performance of HORCRUX with state-of-the-art works FIRE [35], OptML [11], and R2F2 [54]. We evaluated our performance in multiple environments. We implemented these baselines from scratch.

5.1 Microbenchmarks

We provide microbenchmarks to evaluate the performance of HORCRUX and other state-of-the-art systems. Specifically, we trained HORCRUX, FIRE, and OptML using simulated channel measurements. The simulated channels during training are uncorrelated (we used a path loss correlation value of ~0.01). We used 2-antenna and 4-antenna base stations to implement the system for testing. We moved the client across more than 500 different LOS and NLOS positions with varying SNR in both household and office environments. Fig. 13 and Fig. 14 show the prediction performance of HORCRUX. We observe that HORCRUX predictions are very similar to the actual channel measurements (absolute and phase values).

5.1.1 Channel Prediction Accuracy. We defined channel prediction accuracy by Eq. 6. It is an essential metric in the MU-MIMO setup, as errors in predicting the downlink channel can result in interference across clients, affecting SINR performance. Fig. 15 and Fig. 16 show the CDF plot of channel prediction accuracy gain in dB of HORCRUX compared to state-of-the-art systems. Experiments are done in LOS and NLOS conditions for a 2-antenna base station and a single

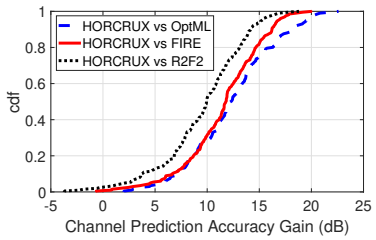


Figure 16: HORCRUX outperforms baselines in NLOS conditions for a 2-antenna base station in a small room environment.

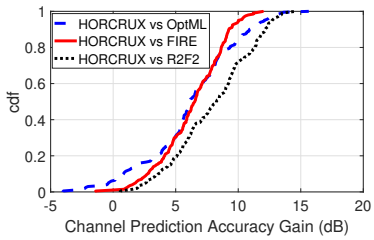


Figure 17: Channel prediction accuracy for a 4-antenna base station. Experiments are performed in a large office environment.

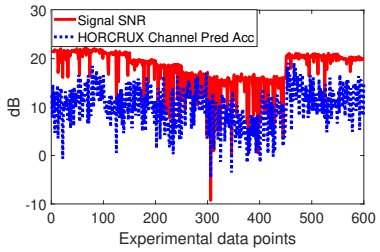


Figure 18: HORCRUX performs accurately under varying SNR. The mean channel prediction accuracy is ~13 dB.

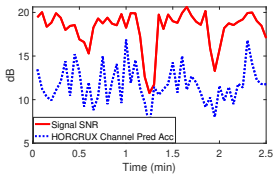


Figure 19: HORCRUX performs accurately for a moving client.

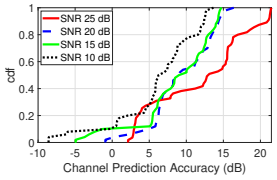


Figure 20: HORCRUX performance varies with signal SNR.

client in a small room. The mean CDF channel prediction accuracy for HORCRUX for LOS condition is 10.34 dB and for NLOS is 11.02 dB. We observe that HORCRUX achieves ~8 dB gain compared to the state-of-the-art systems. The main reason is that FIRE, OptML, and R2F2 perform poorly in multipath-rich environments and need accurate knowledge to train in such environments. Such low prediction accuracy affects MU-MIMO for such systems. However, HORCRUX, because of its unique architecture, can perform accurately in such environments. We reiterate that HORCRUX is trained on simulation data only and has no prior knowledge of the environment.

Fig. 17 shows the performance of HORCRUX using a 4-antenna base station. The experiments are performed on a single client across 200 positions in an office environment. HORCRUX continues to outperform the baselines by a margin of ~6 dB. We note that channel prediction accuracy is dependent on the signal SNR. The red line in Fig. 18 shows the signal SNR across our experimental measurements. The SNR varies between 10-20 dB. The blue dotted line in Fig. 18 shows the HORCRUX channel prediction accuracy for different SNRs. HORCRUX continues to perform quite accurately even under low SNR conditions. We also performed channel prediction for a moving client. We moved a client around the office room for 2.5 minutes and predicted the downlink channel. We observe (see Fig. 19) that HORCRUX performs accurately under varied SNR conditions.

To understand this relationship between accuracy and SNR in detail, we performed controlled experiments where the client was placed in locations with SNR varying from 25 dB to 10 dB. Fig. 20 shows the performance of HORCRUX across different SNR values - we observe that HORCRUX achieves around 15 dB mean channel prediction accuracy for high SNR environments (25 dB). Accuracy decreases with a decrease in SNR values. Such variation is also observed in state-of-the-art systems. However, even in a low SNR environment (10 dB), HORCRUX performs quite accurately.

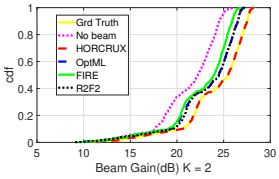


Figure 21: Beamforming gain for a two-antenna base station. HORCRUX achieves 0.1 dB lower than ground truth.

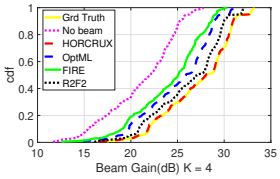


Figure 22: Beamforming gain for a four-antenna base station. Beamforming gain increases with antenna size.

5.1.2 Beamforming Performance. One of the applications of MIMO systems is beamforming, where the base station uses multiple antennas to steer a beam to a specific client. Such techniques can improve SNR and data rates at the client. We evaluate our performance by utilizing the predicted channels in the downlink to beamform to the client. We compare our result to the ground truth channel (Grd Truth) that will achieve optimal beamforming and a random downlink channel as the baseline where there is no beamforming (No beam).

Fig. 21 and Fig. 22 show the performance of HORCRUX for 2-antenna and 4-antenna setups, respectively. As you can see, HORCRUX can beamform near optimally to the ground truth channel beam. The beam gain difference from optimal is < 0.1 dB. FIRE, OptML, and R2F2 follow closely with a beam gain difference of ~2 dB from HORCRUX. We note that the channel accuracy requirement for beam gain performance is low [35]. Thus, state-of-the-art works can perform well in beamforming even with low channel accuracy.

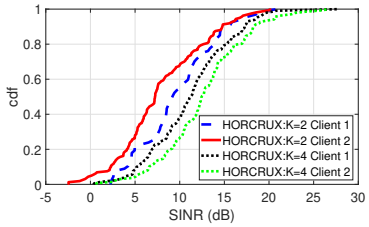


Figure 23: HORCRUX achieves mean ~8 dB MU-MIMO SINR for 2-antenna and 4-antenna base station across two clients.

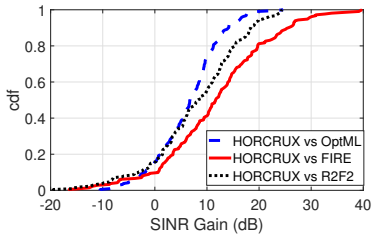


Figure 24: For the 4-antenna setup HORCRUX achieves ~9 dB MU-MIMO SINR gain over state-of-the-art.

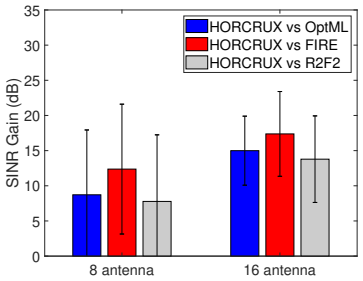


Figure 25: SINR gain across antenna size. HORCRUX achieves > 10 dB SINR gain for 16 antennae

5.1.3 Multi-User MIMO Performance. Multi-antenna base stations enable multi-user MIMO technique where it can simultaneously talk with multiple clients. However, the onerous requirement to perform MU-MIMO is to predict accurate downlink channels. Errors in channel prediction can result in signal leakage into different non-intended clients and cause interference. We use SINR as the metric to evaluate the performance of MU-MIMO. We experimented on 2-antenna and 4-antenna base stations, talking to two clients using the zero-forcing technique mentioned in Sec. 2.2. The experiments are repeated across more than 20 different locations of the clients for office environments with well-conditioned channels. HORCRUX achieves 9.2 dB median SINR for the 2-antenna 2-client setup and 11.12 dB for the 4-antenna 2-client setup. Fig. 23 shows the performance of HORCRUX for 2-antenna and 4-antenna setups for each of the clients. Fig. 24 shows the SINR gain of HORCRUX over state-of-the-art systems for the 4-antenna setup. As you can see, HORCRUX outperforms the baselines by a mean of 8-10 dB. Fig. 25 shows the performance of HORCRUX across larger antenna systems. We used simulation data to evaluate the performance as we could only support a 4-antenna array in our setup. The uplink channels we generated using Eq. 1 and four multipaths ranging from 1 to 200 meters were used. As you can see, for an increase in antenna elements, HORCRUX outperforms the baselines with a higher margin. For 16-antenna elements, HORCRUX has around 15 dB improvement over the state-of-the-art. The main reason is that FIRE, OptML, and R2F2 estimate channels with low accuracy in multipath-rich environments (i.e., uncorrelated observed channel path loss).

5.2 Robustness Across Environment

To evaluate our performance across different conditions, we experimented in 6 different environments: i) Small room, ii) Large Office, iii) Argos indoor, iv) Argos outdoor, v) Simulated environment with four multipaths ranging from 1-100 m, vi) Simulated environment with two multipaths ranging from 1-50 m.

Fig. 29 shows the path loss correlation values in different environments. It measures the relationship of the observed channels in that particular environment. We observe that Small room, Argos indoor, and Argos outdoor experience high mean path correlation (> 0.5) among the channels observed, while Office and the simulated environments show uncorrelated channel observations (< 0.5). A realistic experimental setup should have uncorrelated observations because of the varying multipath combinations. We find that path loss correlation is a key metric that helps explain the comparitive performance of HORCRUX with baselines.

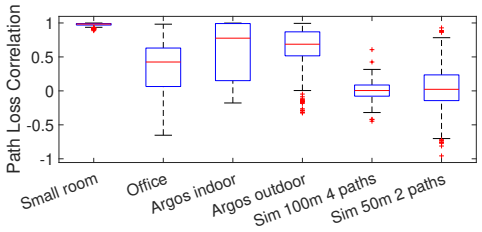


Figure 29: Path loss correlation values for different testbeds. A realistic environment is expected to have a value < 0.5.

The Argos[2] channel measurements were collected in the RENEW testbed [47] with 64-antenna base stations across eight clients. We used a subset of the data (four antennae across four clients) to evaluate our performance. 2-antenna base stations were used in the small room setup, and the client was moved around at 20 different locations a few feet apart. In the office environment, a 4-antenna base station was used, and the client was moved around 20 different locations around 10 feet apart.

We trained FIRE on the Argos indoor dataset and tested it for all the environments. For HORCRUX, we used simulated data as mentioned in Sec. 3 for training. As observed in

[2]We used Argos as FIRE [35] used this testbed for evaluation.

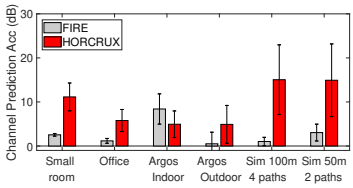


Figure 26: FIRE was trained on Argos indoor and tested across other environments. It can only perform accurately in Argos indoor and fails in all other cases. Thus, FIRE requires to be trained and tested across each environment. However, HORCRUX generalizes accurately across each case without any knowledge about the environment.

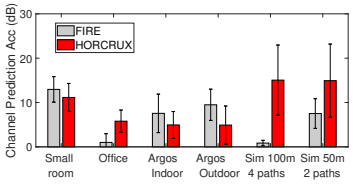


Figure 27: FIRE trained and tested in each of the environments separately. FIRE fails to perform in an environment with uncorrelated channels (HORCRUX office and simulated environments). HORCRUX enjoys good accuracy across all environments. Thus, HORCRUX is robust across wireless environments.

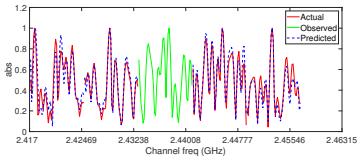


Figure 28: HORCRUX accurately estimates channel across different frequency bands. The green zone signifies the observed channel. FIRE requires to be trained for a particular uplink and downlink frequency pair and does not generalize well across another frequency band.

Fig. 26, FIRE performs well in the trained environment (Argos indoor), but it fails to estimate the channel accurately across other environments. FIRE, being an end-to-end architecture, fails to capture the underlying physical environment and cannot be generalized across other environments having different combinations of multipaths.

We also trained and tested FIRE on each of the environments separately. Fig. 27 shows the performance across multiple environments. As we can see, even with enough knowledge about the wireless environment, FIRE can only perform well in an environment (Small room, Argos indoor and Argos outdoor) with correlated channels, as shown in Fig. 29. In an environment with uncorrelated channels, FIRE fails to perform well.

On the other hand, as seen in Fig. 26 and 27, HORCRUX enjoys accurate estimations across different environments. It is trained only on simulated data and does not require training across environments. Unlike systems trained on specific environmental conditions, the digital twin-based training framework within HORCRUX, with the help of NNCD and mNNDE architectures, as mentioned in Sec. 3, captures a broad range of possible physical environments and thus can give good channel prediction allowing it to generalize to arbitrary environments. However, for the Argos dataset, there is a decrease in channel prediction accuracy because of inaccurate offset correction.

5.3 FDD Hardware Implementation

We implemented an FDD setup using a 2-antenna base station. Uplink and downlink are in different frequency channels, and separate radio-frequency (RF) chains are used. Experiments are carried out in multiple LOS and NLOS positions with varying SNR. We measured uplink on channel 1 (2.412 GHz) and predicted downlink on channel 6 (2.437 GHz - 25 MHz separation) and channel 11 (2.462 GHz - 50 MHz separation), respectively, in a separate set of experiments. Fig. 30 and 31 show the channel prediction accuracy gain of HORCRUX compared to state-of-the-art where uplink-downlink are separated by 25 MHz and 50 MHz, respectively. HORCRUX enjoys a mean gain of ~6 dB. To summarize, HORCRUX outperforms the state-of-the-art on the FDD hardware implementation.

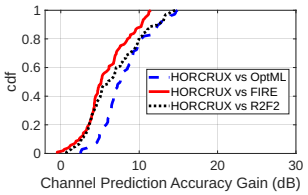


Figure 30: HORCRUX outperforms baseline in an FDD setup with 25 MHz separation between uplink and downlink.

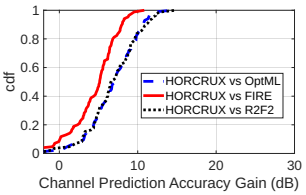


Figure 31: HORCRUX outperforms baseline in an FDD setup with 50 MHz separation between uplink and downlink.

5.4 Across Frequency Bands

To show the performance of HORCRUX across the different frequency bands, we performed a simulation where we observed the uplink band on a particular frequency and tried to predict channel measurements across different frequency bands. Fig. 28 shows the performance of HORCRUX across different downlink frequency bands. The green line shows the observed channel measurements in a particular band. HORCRUX accurately predicted the downlink channel across different bands (see dotted blue line). However, for state-of-the-art FIRE, one must extensively train the system (100K

channel measurements) for separate downlink bands to predict in that particular band, which is time-consuming and not always feasible.

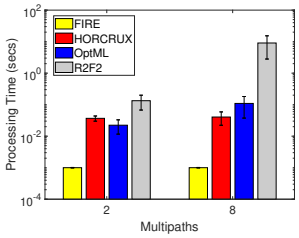


Figure 32: HORCRUX performs competitively better than baselines and 4x and 50x faster than OptML and R2F2 for eight multipath components

5.5 Processing Time

Given the observed uplink channel, an essential performance aspect is the processing time to estimate the downlink channel. Fig. 32 shows the processing time required for state-of-the-art systems. FIRE does not require estimating the underlying physical parameters and thus can estimate the downlink channel within a mean of 1 ms across any number of multipath components. For HORCRUX, OptML, and R2F2, the processing time includes the estimation of initial guesses and the optimization framework to estimate the channel. HORCRUX observes a run time of ∼30 ms for both 2 and 8 multipath components. The increased processing time is mostly because of the optimization framework. However, due to NNCD and mNNDE, HORCRUX can rapidly converge with a small search window. Thus, even with an increase in the number of components, HORCRUX performs almost similarly. OptML performs faster if the number of components is less (∼2), however as shown in Fig. 32, with the increase in the number of multipath components, the optimization framework takes a reasonable amount of time to converge. HORCRUX outperforms OptML by 4x. R2F2 performs worst because of its signal processing technique to estimate the initial guesses. HORCRUX is more than 50x faster than R2F2 in 8 multipath case. The coherence time for the outdoor environment for the 2.4 GHz band is 1-50 ms [35], which means the underlying channel parameters do not change much within this time frame. HORCRUX thus can accurately estimate the downlink channel. Even though FIRE performs the fastest, it cannot be generalized and extended to other environments and frequency bands as it is unable to estimate the underlying physical parameters.

6 RELATED WORKS

6.1 Cross band Channel Prediction

Cross band channel prediction has been well-researched in recent years to eliminate the requirement of feedback overhead in FDD systems. Works like [21, 31] addressed the feedback issue by compressing the overhead required. However, these techniques are not zero-feedback-based systems like our proposed work. Zero-feedback-based approaches are also introduced in works like [11, 26, 27, 39, 43, 54, 59]. Most of these works utilize signal processing or a physics-based approach to estimate the downlink channels. However, these techniques suffer from excessive processing time and fail to perform accurately in multipath-rich environments. Works like [7, 9, 11, 35, 44, 46, 50, 56, 64, 65] uses different machine learning architecture like convoluted neural network (CNN), feedforward neural network (FNN) and variational autoencoders (VAE). However, these works are end-to-end systems that predict the downlink directly from the uplink channel. These techniques require very small processing time for prediction, however, they require to be trained on the particular environment and frequency band and cannot be generalized. Unlike these works, HORCRUX introduces physics-based machine learning architectures as mentioned in Sec. 3 that can estimate the underlying physical parameters and thus can estimate the downlink channel across environments and frequency without any knowledge of the wireless environment.

6.2 Machine Learning in Wireless Systems

Recent years have seen considerable development in using machine learning in wireless system designs. ML-aided wireless sensing [2, 3, 8, 32], localization [4, 14, 24, 40], in-band channel prediction and modeling [42, 62, 66, 67], MIMO systems [17, 18, 23, 51] have shown great improvement. Our proposed work follows the same vision, however, it utilizes machine learning to retrieve the physical information of a wireless channel. Such fine-grained information can help in various other domains of research in wireless systems. Mixture of Experts (MoE) is a machine-learning technique that has been used in recent years [13, 45, 48, 55, 68] to divide a problem into sub-tasks, each managed by an expert (NN). A gating model is normally used to select which expert to trust while generating predictions. Such technique has been proven efficient in vision applications for fine-grained training and achieving super-resolution. HORCRUX follows a similar divide-and-conquer approach to predict the downlink channel, however, our technique is different from MoE's as we do not require any gating mechanism and each of the neural networks is trained independently. Such flexibility makes our system easy to train and scale.

7 DISCUSSION

1) Delay spread: In wireless communications, the delay spread of a channel refers to the difference in path length of the first and the last path in the channel. Thus, the range

of the delay spread governs the number of possible multipath components, which in turn corresponds to the size of the NNCD and mNNDE used within HORCRUX and the amount of dataset required to train. Throughout this paper, we trained HORCRUX to resolve multipath within the delay spread range of 0-200 m, which is a realistic assumption and has been used in related works [11]. If the distance traveled by the multipath components exceeds 200 m, it will not be identified by HORCRUX, resulting in erroneous channel prediction. To remedy that, we will use the shifting vector technique introduced by OptML [11]. We can get very coarse estimates of the distances from the IFFT bin of the channel impulse response, which can be used to compute the d_{shift} (shifting the distance estimates within the trained delay spread). Such a technique can restrict the training for NNCD and mNNDE to the maximum delay spread of 200 m.

2) Sampling rate or Signal bandwidth: The sampling rate or signal bandwidth determines the ability to resolve two multipath components arriving at different times. Thus, a 10 MHz bandwidth limits our resolution to 30 m, i.e., one can resolve multipaths traveling at least 30 m apart, while a 20 MHz bandwidth allows a finer resolution of 15 m. However, if paths arrive within that resolution range, it becomes difficult for our system to resolve them. HORCRUX will estimate some composite distance. Such an estimation can affect the channel prediction accuracy. However, all physics-based systems [11, 54] will suffer from this limitation. Successive interference cancellation techniques [20] is one way to counter this limitation, as it can be used to estimate the paths iteratively.

3) Hardware imperfections: The uplink channel at the base station is measured using the preambles transmitted by the client. Such channel measurements are associated with CFO, SFO, and hardware imperfection delays. Such offsets can be measured and corrected by established techniques [35, 41]. It is to be noted that such corrections are essential for HORCRUX, or else it can result in erroneous distance estimates.

4) Processing time minimization: Processing time or run time is the time to predict the downlink channel from the observed uplink. HORCRUX takes around 30 ms to estimate the channel. It is faster than most state-of-the-art systems but slower than end-to-end architectures like FIRE. Much of the processing time for HORCRUX is due to the optimization framework. However, such a framework can be further optimized using local optimization. At this point, HORCRUX uses all the distance estimates together to fit the uplink channel as mentioned in Sec. 3.4. We can further subdivide this optimization framework where the distance estimate from each $mNNDE_i$ can be used to fit into the output of $NNCD_i$ locally. Such a design can decrease the optimization time by 10x. However, for such a design to work, NNCD blocks must estimate with high prediction accuracy. This research improvement has been kept for future work. Throughout this paper, our primary goal was to design a system that can be generalized across environments with high accuracy and reasonable processing time.

5) Prediction across space: We note that each antenna element runs HORCRUX parallelly. The size of the HORCRUX model operating on a single antenna is ~7 MB. Thus, the model's size will increase linearly for a massive MIMO system with 64-128 antenna elements. However, the path distance of multipath components arriving at each antenna will differ based on the architecture of the antenna array. Thus, if the antenna array architecture is known, HORCRUX can use the distance estimates for one antenna in the optimization framework to predict the downlink channel across other antenna elements in the array. We have left this work for the future.

8 CONCLUSION

In this paper, we present HORCRUX, a physics-based machine learning system that can predict wireless channels across different frequency bands and can be used by various wireless devices ranging from single-antenna IoT devices to massive MIMO base stations. We evaluated and compared our system with state-of-the-art techniques across different wireless testbeds and simulations. Our evaluations show that HORCRUX can predict crossband channels with high accuracy and achieves ~8 dB SINR gain over the state-of-the-art. To the best of our knowledge, HORCRUX is a first-of-its-kind system that can continue to provide accurate estimations across multiple wireless environments with low correlation without any prior knowledge of that particular environment. Integration of our proposed system will be essential to the success of next-generation wireless networks, such as 5G and 6G, and these tools will help to improve the overall performance and efficiency of wireless communication systems.

ACKNOWLEDGMENTS

We thank the anonymous reviewers, shepherd, and members of the CoSyNe Communication Systems and Networking group for their feedback on the paper. This material is based upon work supported by the National Science Foundation (NSF) under grant numbers: 2007581, 2112471, and 2128567. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] *WARP Project.* http://warpproject.org.

[2] Adesina, D., Hsieh, C.-C., Sagduyu, Y. E., and Qian, L. Adversarial machine learning in wireless communications using rf data: A review.

IEEE Communications Surveys & Tutorials (2022).

[3] Ahmad, R., Wazirali, R., and Abu-Ain, T. Machine learning for wireless sensor networks security: An overview of challenges and issues. *Sensors 22*, 13 (2022), 4730.

[4] Akhil, K., and Sinha, S. Self-localization in large scale wireless sensor network using machine learning. In *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)* (2020), IEEE, pp. 1–5.

[5] Alamouti, S., Casas, E. F., Hirano, M., Hoole, E., Jesse, M., Michelson, D. G., Poon, P., Veintimilla, G. J., and Zhang, H. Method for frequency division duplex communications, Aug. 3 1999. US Patent 5,933,421.

[6] Alexopoulos, K., Nikolakis, N., and Chryssolouris, G. Digital twin-driven supervised machine learning for the development of artificial intelligence applications in manufacturing. *International Journal of Computer Integrated Manufacturing 33*, 5 (2020), 429–439.

[7] Alrabeiah, M., and Alkhateeb, A. Deep learning for tdd and fdd massive mimo: Mapping channels in space and frequency. In *2019 53rd asilomar conference on signals, systems, and computers* (2019), IEEE, pp. 1465–1470.

[8] Alsheikh, M. A., Lin, S., Niyato, D., and Tan, H.-P. Machine learning in wireless sensor networks: Algorithms, strategies, and applications. *IEEE Communications Surveys & Tutorials 16*, 4 (2014), 1996–2018.

[9] Arnold, M., Dörner, S., Cammerer, S., Yan, S., Hoydis, J., and Brink, S. t. Enabling fdd massive mimo through deep learning-based channel prediction. *arXiv preprint arXiv:1901.03664* (2019).

[10] Bajaj, G., Parthasarathy, S., Shalin, V., and Sheth, A. P. Grounding from an artificial intelligence and cognitive science lens. *IEEE Intelligent Systems* (2024).

[11] Bakshi, A., Mao, Y., Srinivasan, K., and Parthasarathy, S. Fast and efficient cross band channel prediction using machine learning. In *The 25th Annual International Conference on Mobile Computing and Networking* (2019), pp. 1–16.

[12] Björnson, E., Larsson, E. G., and Marzetta, T. L. Massive mimo: Ten myths and one critical question. *IEEE Communications Magazine 54*, 2 (2016), 114–123.

[13] Brachmann, E., and Rother, C. Expert sample consensus applied to camera re-localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 7525–7534.

[14] Burghal, D., Ravi, A. T., Rao, V., Alghafis, A. A., and Molisch, A. F. A comprehensive survey of machine learning based localization with wireless signals. *arXiv preprint arXiv:2012.11171* (2020).

[15] Choi, J., Love, D. J., and Bidigare, P. Downlink training techniques for fdd massive mimo systems: Open-loop and closed-loop training with memory. *IEEE Journal of Selected Topics in Signal Processing 8*, 5 (2014), 802–814.

[16] Cohen, I., Huang, Y., Chen, J., Benesty, J., Benesty, J., Chen, J., Huang, Y., and Cohen, I. Pearson correlation coefficient. *Noise reduction in speech processing* (2009), 1–4.

[17] Gecgel, S., Goztepe, C., and Kurt, G. K. Transmit antenna selection for large-scale mimo gsm with machine learning. *IEEE Wireless Communications Letters 9*, 1 (2019), 113–116.

[18] Goutay, M., Aoudia, F. A., Hoydis, J., and Gorce, J.-M. Machine learning for mu-mimo receive processing in ofdm systems. *IEEE Journal on Selected Areas in Communications 39*, 8 (2021), 2318–2332.

[19] Gulli, A., and Pal, S. *Deep learning with Keras*. Packt Publishing Ltd, 2017.

[20] Halperin, D., Anderson, T., and Wetherall, D. Taking the sting out of carrier sense: interference cancellation for wireless lans. In *Proceedings of the 14th ACM international conference on Mobile computing and networking* (2008), pp. 339–350.

[21] Han, Y., Hsu, T.-H., Wen, C.-K., Wong, K.-K., and Jin, S. Efficient downlink channel reconstruction for fdd transmission systems. In *2018 27th Wireless and Optical Communication Conference (WOCC)* (2018), IEEE, pp. 1–5.

[22] Hardesty, L. T-mobile exec touts massive mimo for both tdd and fdd bands. *Fierce Wireless* (2020).

[23] He, H., Zhang, M., Jin, S., Wen, C.-K., and Li, G. Y. Model-driven deep learning for massive mu-mimo with finite-alphabet precoding. *IEEE Communications Letters 24*, 10 (2020), 2216–2220.

[24] Helal, S., Sarieddeen, H., Dahrouj, H., Al-Naffouri, T. Y., and Alouini, M.-S. Signal processing and machine learning techniques for terahertz sensing: An overview. *IEEE Signal Processing Magazine 39*, 5 (2022), 42–62.

[25] Hu, C., Fan, W., Zeng, E., Hang, Z., Wang, F., Qi, L., and Bhuiyan, M. Z. A. Digital twin-assisted real-time traffic data prediction method for 5g-enabled internet of vehicles. *IEEE Transactions on Industrial Informatics 18*, 4 (2021), 2811–2819.

[26] Huang, C., Alexandropoulos, G. C., Zappone, A., Yuen, C., and Debbah, M. Deep learning for ul/dl channel calibration in generic massive mimo systems. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)* (2019), IEEE, pp. 1–6.

[27] Hugl, K., Kalliola, K., Laurila, J., et al. Spatial reciprocity of uplink and downlink radio channels in fdd systems. In *Proc. COST* (2002), vol. 273, Citeseer, p. 066.

[28] Irmer, R., Droste, H., Marsch, P., Grieger, M., Fettweis, G., Brueck, S., Mayer, H.-P., Thiele, L., and Jungnickel, V. Coordinated multipoint: Concepts, performance, and field trial results. *IEEE Communications Magazine 49*, 2 (2011), 102–111.

[29] Jones, E., Oliphant, T., and Peterson, P. others.(2001). scipy: Open source scientific tools for python, 2014.

[30] Katabi, H. Megamimo: Scaling wireless capacity with user demands.

[31] Khalilsarai, M. B., Haghighatshoar, S., Yi, X., and Caire, G. Fdd massive mimo: Efficient downlink probing and uplink feedback via active channel sparsification. In *2018 IEEE International Conference on Communications (ICC)* (2018), IEEE, pp. 1–6.

[32] Khan, Z. A., and Samad, A. A study of machine learning in wireless sensor network. *Int. J. Comput. Netw. Appl 4*, 4 (2017), 105–112.

[33] Liang, L., Xu, W., and Dong, X. Low-complexity hybrid precoding in massive multiuser mimo systems. *IEEE Wireless Communications Letters 3*, 6 (2014), 653–656.

[34] Liu, L., Chen, R., Geirhofer, S., Sayana, K., Shi, Z., and Zhou, Y. Downlink mimo in lte-advanced: Su-mimo vs. mu-mimo. *IEEE Communications Magazine 50*, 2 (2012), 140–147.

[35] Liu, Z., Singh, G., Xu, C., and Vasisht, D. Fire: enabling reciprocity for fdd mimo systems. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking* (2021), pp. 628–641.

[36] Lu, L., Li, G. Y., Swindlehurst, A. L., Ashikhmin, A., and Zhang, R. An overview of massive mimo: Benefits and challenges. *IEEE journal of selected topics in signal processing 8*, 5 (2014), 742–758.

[37] Marzetta, T. L., and Ngo, H. Q. *Fundamentals of massive MIMO*. Cambridge University Press, 2016.

[38] McClellan, A., Lorenzetti, J., Pavone, M., and Farhat, C. A physics-based digital twin for model predictive control of autonomous unmanned aerial vehicle landing. *Philosophical Transactions of the Royal Society A 380, 2229* (2022), 20210204.

[39] Min, C., Chang, N., Cha, J., and Kang, J. Mimo-ofdm downlink channel prediction for ieee802. 16e systems using kalman filter. In *2007 IEEE Wireless Communications and Networking Conference* (2007), IEEE, pp. 942–946.

[40] Nguyen, C. L., Georgiou, O., and Gradoni, G. Reconfigurable intelligent surfaces and machine learning for wireless fingerprinting localization. *arXiv preprint arXiv:2010.03251* (2020).

[41] Nishad, P. K., and Singh, P. Carrier frequency offset estimation in

ofdm systems. In *2013 IEEE Conference on Information & Communication Technologies* (2013), IEEE, pp. 885–889.

[42] O'Shea, T. J., Roy, T., and West, N. Approximating the void: Learning stochastic channel models from observation with variational generative adversarial networks. In *2019 International Conference on Computing, Networking and Communications (ICNC)* (2019), IEEE, pp. 681–686.

[43] Peng, W., Li, W., Wang, W., Wei, X., and Jiang, T. Downlink channel prediction for time-varying fdd massive mimo systems. *IEEE Journal of Selected Topics in Signal Processing 13*, 5 (2019), 1090–1102.

[44] Pérez Gómez, A. Ml-aided cross-band channel prediction in mimo systems, 2022.

[45] Riquelme, C., Puigcerver, J., Mustafa, B., Neumann, M., Jenatton, R., Susano Pinto, A., Keysers, D., and Houlsby, N. Scaling vision with sparse mixture of experts. *Advances in Neural Information Processing Systems 34* (2021), 8583–8595.

[46] Safari, M. S., Pourahmadi, V., and Sodagari, S. Deep ul2dl: Data-driven channel knowledge transfer from uplink to downlink. *IEEE Open Journal of Vehicular Technology 1* (2019), 29–44.

[47] Shepard, C., Ding, J., Guerra, R. E., and Zhong, L. Understanding real many-antenna mu-mimo channels. 461–467.

[48] Shi, Y., Paige, B., Torr, P., et al. Variational mixture-of-experts autoencoders for multi-modal deep generative models. *Advances in Neural Information Processing Systems 32* (2019).

[49] Shuang, T., Koivisto, T., Maattanen, H.-L., Pietikainen, K., Roman, T., and Enescu, M. Design and evaluation of lte-advanced double codebook. In *2011 IEEE 73rd Vehicular Technology Conference (VTC Spring)* (2011), IEEE, pp. 1–5.

[50] Soltani, M., Pourahmadi, V., Mirzaei, A., and Sheikhzadeh, H. Deep learning-based channel estimation. *IEEE Communications Letters 23*, 4 (2019), 652–655.

[51] Song, J., Häger, C., Schröder, J., O'Shea, T., and Wymeersch, H. Benchmarking end-to-end learning of mimo physical-layer communication. In *GLOBECOM 2020-2020 IEEE Global Communications Conference* (2020), IEEE, pp. 1–6.

[52] Tse, D., and Viswanath, P. *Fundamentals of wireless communication*. Cambridge university press, 2005.

[53] Van Der Walt, S., Colbert, S. C., and Varoquaux, G. The numpy array: a structure for efficient numerical computation. *Computing in science & engineering 13*, 2 (2011), 22–30.

[54] Vasisht, D., Kumar, S., Rahul, H., and Katabi, D. Eliminating channel feedback in next-generation cellular networks. In *Proceedings of the 2016 ACM SIGCOMM Conference* (2016), pp. 398–411.

[55] Wang, D., Mo, J., Zhou, G., Xu, L., and Liu, Y. An efficient mixture of deep and machine learning models for covid-19 diagnosis in chest x-ray images. *PloS one 15*, 11 (2020), e0242535.

[56] Wang, J., Ding, Y., Bian, S., Peng, Y., Liu, M., and Gui, G. Ul-csi data driven deep learning for predicting dl-csi in cellular fdd systems. *IEEE Access 7* (2019), 96105–96112.

[57] Wiesel, A., Eldar, Y. C., and Shamai, S. Zero-forcing precoding and generalized inverses. *IEEE Transactions on Signal Processing 56*, 9 (2008), 4409–4418.

[58] Wu, L., Chen, J., Yang, H., and Lu, D. Codebook design for cross-polarized linear antenna array in lte-a downlink system. In *2011 IEEE Vehicular Technology Conference (VTC Fall)* (2011), IEEE, pp. 1–5.

[59] Xie, Y., Xiong, J., Li, M., and Jamieson, K. md-track: Leveraging multi-dimensionality for passive indoor wi-fi tracking. In *The 25th Annual International Conference on Mobile Computing and Networking* (2019), pp. 1–16.

[60] Xu, Y., Yue, G., and Mao, S. User grouping for massive mimo in fdd systems: New design methods and analysis. *IEEE Access 2* (2014), 947–959.

[61] Yacoby, Y., Pan, W., and Doshi-Velez, F. Failure modes of variational autoencoders and their effects on downstream tasks. *arXiv preprint arXiv:2007.07124* (2020).

[62] Yang, G., Zhang, Y., He, Z., Wen, J., Ji, Z., and Li, Y. Machine-learning-based prediction methods for path loss and delay spread in air-to-ground millimetre-wave channels. *IET Microwaves, Antennas & Propagation 13*, 8 (2019), 1113–1121.

[63] Yang, Q., Li, X., Yao, H., Fang, J., Tan, K., Hu, W., Zhang, J., and Zhang, Y. Bigstation: Enabling scalable real-time signal processingin large mu-mimo systems. *ACM SIGCOMM Computer Communication Review 43*, 4 (2013), 399–410.

[64] Yang, Y., Gao, F., Li, G. Y., and Jian, M. Deep learning-based downlink channel prediction for fdd massive mimo system. *IEEE Communications Letters 23*, 11 (2019), 1994–1998.

[65] Yang, Y., Gao, F., Zhong, Z., Ai, B., and Alkhateeb, A. Deep transfer learning-based downlink channel prediction for fdd massive mimo systems. *IEEE Transactions on Communications 68*, 12 (2020), 7485–7497.

[66] Yang, Y., Li, Y., Zhang, W., Qin, F., Zhu, P., and Wang, C.-X. Generative-adversarial-network-based wireless channel modeling: Challenges and opportunities. *IEEE Communications Magazine 57*, 3 (2019), 22–27.

[67] Yuan, J., Ngo, H. Q., and Matthaiou, M. Machine learning-based channel prediction in massive mimo with channel aging. *IEEE Transactions on Wireless Communications 19*, 5 (2020), 2960–2973.

[68] Zhang, L., Huang, S., Liu, W., and Tao, D. Learning a mixture of granularity-specific experts for fine-grained categorization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), pp. 8331–8340.

[69] Zhou, X., Xu, X., Liang, W., Zeng, Z., Shimizu, S., Yang, L. T., and Jin, Q. Intelligent small object detection for digital twin in smart manufacturing with industrial cyber-physical systems. *IEEE Transactions on Industrial Informatics 18*, 2 (2021), 1377–1386.