

AirFC: Designing Fully Connected Layers for Neural Networks with Wireless Signals

Guillem Reus-Muns*, Kubra Alemdar*, Sara Garcia Sanchez, Debashri Roy, Kaushik R. Chowdhury
Institute for the Wireless Internet of Things, Northeastern University, Boston, MA, USA
{greusmuns,kalemdar,sgarcia}@coe.neu.edu,{droy,krc}@ece.neu.edu

ABSTRACT

This paper proposes and experimentally validates a new paradigm for computing with wireless signals over-the-air (OTA). It demonstrates the first fully connected (FC) neural network (NN) constructed entirely using channel propagation and signal interference principles. Our design is based on architecting the desired linear operation of an FC layer through the superposition of signals emitted from multiple transmitters and received at a single receiver, similar to multiple input single output (MISO) systems. Our design takes into account several practical considerations, such as the impact of multiple subcarriers, the number of transmit antennas, and the changing wireless channel. The key outcome of our work is developing a principled methodology that transforms a given trained digital FC NN into its OTA equivalent. This novel computational paradigm, which we call AirFC, allows us to run NN tasks without compute-specific hardware during tests. We validate our design using 9 time-synchronized software-defined radios (SDRs) available on the ORBIT testbed, emulating a 16 antenna array. We use the MNIST dataset as input to our wireless FC NN and demonstrate classification with 92.61% accuracy, which proves that our NN with OTA FC layers performs similar to the conventional, all-digital version with an accuracy decrease of only 0.73%.

CCS CONCEPTS

• Computing methodologies → Neural networks; • Networks → Network experimentation.

KEYWORDS

over-the-air, analog computation, fully connected layer, neural networks, multiple-input single-output channels

ACM Reference Format:

Guillem Reus-Muns*, Kubra Alemdar*, Sara Garcia Sanchez, Debashri Roy, Kaushik R. Chowdhury. 2023. AirFC: Designing Fully Connected Layers for Neural Networks with Wireless Signals. In *International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing (MobiHoc '23)*, October 23–26, 2023, Washington, DC, USA. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3565287.3610281>

*G. Reus-Muns and K. Alemdar have equally contributed to this work.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MobiHoc '23, October 23–26, 2023, Washington, DC, USA
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-9926-5/23/10...\$15.00
<https://doi.org/10.1145/3565287.3610281>

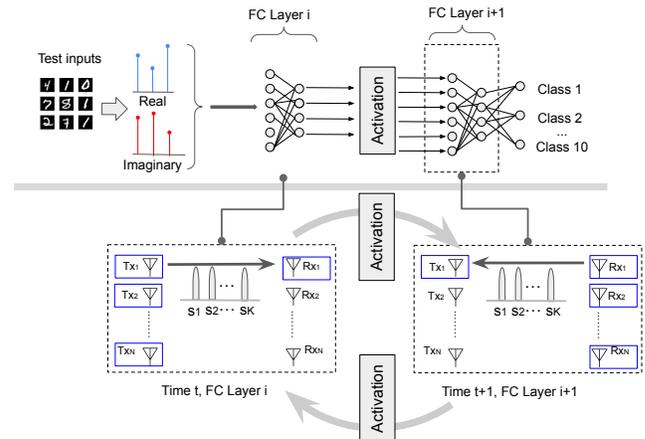


Figure 1: AirFC architecture shows the FC operation performed over-the-air using a MIMO system. N transmitters are used to send N complex data samples. The output at each neuron is obtained using different OFDM sub-carriers and a single receiver. The MIMO architecture allows to switch the roles of transmitters and receivers to replicate the input and output of the FC layer without a feedback loop.

1 INTRODUCTION

Radio Frequency (RF) signals are being increasingly used for exciting applications that go beyond traditional communications, such as sensing, imaging and positioning. As illustrative examples of this paradigm shift that uses classical communications systems and links for alternative applications, massive MIMO and re-configurable intelligent surfaces (RIS) have enabled activity/gesture recognition [1], digital twin of propagation environment [2], object detection [3], and others [4, 5]. Along this direction, we specifically highlight the possibility of joint *communications* and *computing*, which forms the main focus of this work. While early research has shown great promise in simultaneously transmitting and averaging data from field-deployed sensors to a central aggregation point [6], we propose and demonstrate experimentally a fundamentally new form of *over-the-air* (OTA) analog computation. Going beyond the state-of-the-art, our approach enables operations required for a neural network using OTA wireless signals.

1.1 Concept of Wireless Analog Computation

OTA computation aims to shift the processing complexity to the analog (wireless) domain by taking advantage of channel transformations and engineering precisely controlled signal reflections and interference, with overall aim of emulating a specific mathematical operation. Recent works have explored OTA computation in the

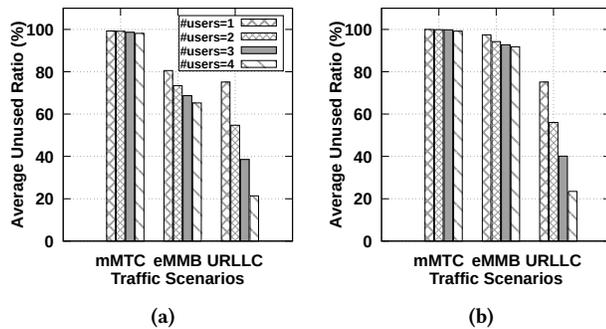


Figure 2: The average percentage of unused PRBs for a fixed time duration when users are; (a) Constantly active, and (b) Randomly active.

context of wireless data aggregation [7] and OTA federated learning [8] (see sec. 8). However, these works are limited to analytical results and validation via simulations. Without a systems-level validation, many practical challenges that occur in actual deployment conditions, including the impact of the channel and the limitations of hardware are not comprehensively considered.

1.2 Building Wireless Neural Networks

We propose harnessing the capability of OTA computation to run inference on a neural network (NN) consisting of a set of fully connected layers (FC). To do so, we design AirFC, the first OTA computation-based neural network that is equivalent to its all-digital counterpart. In general, FC layers are composed of a set of neurons that compute linear operations (weighted sum) followed by a non-linear operation or activation function. In AirFC, we model the weighted sum as a superposition of signals that interfere at the receiver and replicate the linear operation of an FC layer.

Fig. 1 shows the AirFC architecture performing the FC operation over-the-air using a MIMO system, where the number of neurons in each layer is mapped to the equivalent number of transmit antennas in the MIMO architecture, and each transmit stream uses OFDM sub-carriers to perform parallel computation of the different output layer neurons. Therefore, AirFC can compute a matrix-vector multiplication of one layer in one single time slot (i.e., in a symbol time).

1.3 Challenges in AirFC

Designing and implementing AirFC poses a number of challenges:

- **Error characterization:** Inaccuracies in channel estimation, equalization and the receiver thermal noise adversely affect the OTA inference. We first characterize these distortions through empirical measurements using software defined radios. Given that the channel estimation error highly depends on the properties of the training signal, aside of the channel conditions, we investigate the error performance for different training sequence configurations.
- **System synchronization:** In order to achieve coherent reception and the desired interference at the receiver, all system components, including host machines and radios require both time sample-level and frequency synchronization. For instance, in Fig. 1, all transmitters are assumed to be synchronized. This enables to have all radios transmit at the same time, with a time synchronization error smaller than a symbol time (which is μs -level in our implementation).

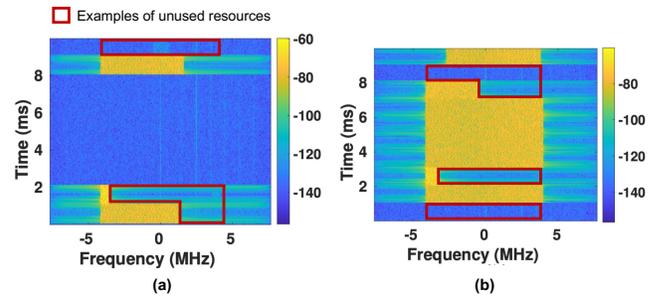


Figure 3: Spectrograms generated out of srsRAN OTA captures. PHY resources are not uniformly used throughout the time-frequency axes, and this usage depends on different traffic conditions.

- **Beamforming:** Multiple antennas will synchronously work to enable coherent transmissions. Thus, system level synchronization is required to implement zero-forcing pre-equalization, which includes channel estimation, channel feedback and beamforming weights computation. Zero-forcing beamforming helps to compensate for the non-uniform fading between multiple antennas.
- **Complex-valued NN:** Wireless signals are represented using complex numbers that include both amplitude and phase information. As shown in Fig. 1, the input data, as well as the weights, are represented using complex values. Thus, we need to design a complex-valued FC layer to fully exploit the unique wireless properties, over the real-valued equivalent NN model.
- **Array-inspired NN:** AirFC relies on leveraging the underlying and pre-existing infrastructure of the network, in terms of available radio/network resources. Thus, the FC layer computation should be designed considering jointly the system limitations and test accuracy of the NN, and this include optimizing the outcome based on number of antennas and number of neurons in each layer of the model.

1.4 Contributions

Our main contributions are as follows:

- We develop and validate the theory that maps digital fully connected layers and analog over-the-air interference.
- We derive an analytical framework to characterize the channel and receiver noises under both perfect channel knowledge and with channel estimation errors.
- We consider realistic limitations on number of available antennas and propose a re-transmission based approach to overcome such hardware limitations.
- As a systems contribution, we demonstrate inference over-the-air using the ORBIT test-bed composed of SDRs.
- We compare the performance of a single layer and end-to-end inference over-the-air with a fully digital implementation, resulting with a 92.61% accuracy on the MNIST dataset, which represents only a 0.73% reduction in comparison to the all-digital counterpart.

2 USE CASES OF AIRFC

- **High frequency-band UM-MIMO arrays:** 5G/6G cellular communications will exploit higher frequencies in the spectrum, including mmWave and reaching up to sub-THz. Given that antenna dimensions scale proportionally to its wavelength, large arrays with

sub-millimeter antennas will be employed in communications systems in this extremely high frequencies. This will facilitate integrating large numbers of antennas (i.e. 1024x1024 [9]) in nano-antenna arrays, usually referred to as ultra massive MIMO (UM-MIMO) systems. Given that AirFC leverages multi-antenna systems, the availability of large arrays opens up opportunistic scenarios where AirFC can coexist with regular communications, while benefiting from such antenna array that allows higher number of neurons in the FC layer.

- **AirFC on-chip:** Another enticing application is *wireless network on-chip communication* (WiNoC) where mmWave/THz band signals can enable inter-core communication among processors embedded on-chip by utilizing nano-scale antenna arrays [10]. Recent studies have reported that mmWave-based WiNoC outperforms conventional wired NoC architectures in terms of achievable band-width and energy dissipation [11]. This increases the possibility of designing metasurface-based on-chip antenna [12], which will present many opportunities to implement AirFC at a chip-level.

- **Shared Resources for Cellular Networks:** Cellular deployments are becoming dense and the large body of work on network slicing suggests the potential of sharing of resources among nodes. AirFC can reuse existing deployments without requiring any dedicated hardware. Thus, we envision AirFC as an add-on capability to existing and future wireless networks, where resources can be shared between other technologies based on demand.

2.1 AirFC in Practical Cellular Networks

AirFC can coexist within an existing OFDM communication system. In this section, we show how to integrate it within a standard 4G/5G PHY frame structure. This frame structure is divided into discrete resource elements and grouped into physical resource blocks (PRBs), which are then assigned to cellular users based on channel conditions and traffic demands [13]. As reported in [14], not all PRBs are used at all times (see Fig. 3), even after fulfilling all users' traffic demands. This represents an opportunity for leveraging the unused subcarriers for AirFC when traffic demand is low.

2.1.1 Experimental Study. We use the Colosseum emulator [15] to run srsRAN-based (open software for mobile wireless networks [16]) scenarios to analyze cellular resource usage under different traffic loads. In particular, we use a base station (eNB) serving up to 4 users and 3 different types of cellular traffic, Enhanced Mobile Broadband (eMBB), Massive Machine-Type Communications (mMTC) and Ultra-High Reliability & Low Latency (URLLC). The eNB divides the available spectrum into three slices, with each slice having a total of 50 PRBs assigned as radio resources. To allocate resources among active users in each slice, srsRAN utilizes a round-robin scheduling algorithm. Fig. 2 shows the average percentage of unused PRBs for a fixed time duration, and for different traffic types and number of users. Additionally, we distinguish between two different user activities: constantly active and randomly active over time. While at different levels, we observe that spectrum resources remain unused in the range 23.5-99.94% considering all scenarios.

3 SYSTEM MODEL OF AIRFC

Consider a wireless system with a single receiver and a different transmitter with a total of N antennas, or conversely, N transmitters each with a single antenna that are synchronized. From Sec. 1, our

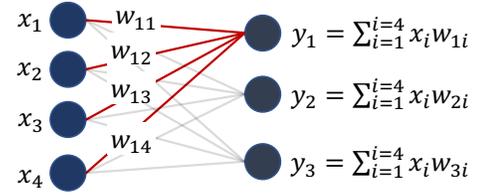


Figure 4: Edges in a FC layers connecting every input neuron to every output neuron are allocated a weight. The output of every neuron prior to the activation function is the weighted sum of the input x and the weights w .

goal is to replicate the linear function that is performed by a fully connected (FC) layer widely used in neural networks.

3.1 Over-The-Air Fully Connected Layers

In this section, we explain the theoretical formulation of the equivalent relationship between the two diverse concepts of (i) FC neural networks and (ii) classical propagation and interference effects.

The parameters (interchangeably used as ‘weights’) of a FC layer are defined as set of numerical weights that connect every input neuron with every output neuron. As shown in Fig. 4, every input value x_i is multiplied by the weight w_{ij} connecting to the output neuron y_j . All the weighted input values ($x_i w_{ij}$) are summed as:

$$y_j = x_1 * w_{j1} + x_2 * w_{j2} + \dots + x_L * w_{jL} = \sum_{i=1}^L x_i * w_{ij} \quad (1)$$

Given that wireless signals and channels can be represented with magnitude and phase, let x, w and y be complex valued scalars. As we describe in Sec. 6.3.1, this enables us to design complex-valued neural networks, which may have the added benefit of higher performance than their real-valued equivalent [17].

- **Over-the-air FC computation:** Our goal is to artificially construct a series of signal transformations with the physical environment that will jointly combine at the receiver. In particular, given a sequence \mathbf{x} of length L and a set of N transmitters, every transmitter i will transmit a symbol x_i in specific time-steps. The remainder of this section assumes that the number of transmitters is equal or greater than L ($N \geq L$). We analyze the particular case of $N < L$ in Sec. 5. Without loss of generality, we consider that the sequence \mathbf{x} as the desired input to a neural network. Then, the received signal r can be expressed as:

$$r = \sum_{i=1}^L p_i h_i x_i + N_z, \quad (2)$$

where r is the received signal, h_i is the flat-fading channel coefficient from transmitter i to the receiver, p_i is a pre-equalization term applied on the transmitter side and N_z is the receiver noise, modeled as a random variable that follows a Gaussian distribution with mean 0 and variance σ_z^2 . Assuming that the channel state information (CSI) is known at every transmitter, the coefficient p_i is computed such that the product $p_i h_i$ is equal to a desired neural network weight w_i , ($r = \sum_{i=1}^L w_i x_i + z$). Thus, the expression for the FC neuron output y is equivalent to the received signal r with AWGN and pre-coding weights $p = w \frac{\hat{h}^*}{\|\hat{h}\|^2}$, with \hat{h}^* given by the complex conjugate of the estimated CSI. Notice that in this formulation, p acts as a zero-forcing equalizer, but the working principle is totally independent of the equalization algorithm employed.

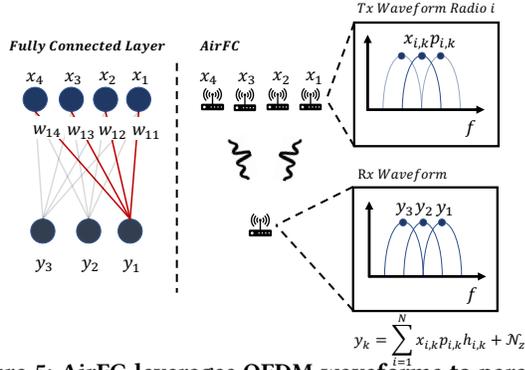


Figure 5: AirFC leverages OFDM waveforms to parallelize different weighted sums over the air. Every subcarrier represents the output to a different neuron (y_k) of FC layer.

• **OFDM parallelization:** The formulation presented up until now only considered single carrier waveforms. Here, we leverage multi-carrier schemes (OFDM) as a technique to parallelize multiple mathematical operations, as shown in Fig. 5. In particular, we aim to use each different subcarrier to represent an independent weighted sum, mapping to a different output neuron. Then, for a given FC layer with N input neurons and M output neurons, every subcarrier k is used to independently compute the output of one of the M neurons. Thus, the received signal can be expressed as:

$$r_k = \sum_{i=1}^L p_{i,k} h_{i,k} x_{i,k} + N_z, \quad (3)$$

where k represents the subcarrier index. Note that this approach does not necessarily assume that the number of available subcarriers is always greater than M , and OFDM is simply proposed as an additional parallelization technique that reduces by a factor of K (number of subcarriers) the number of required transmissions.

3.2 End-to-End System Architecture

• **Architecture overview.** We utilize a MISO (Multiple-Input Single-Output) system to implement AirFC. Note that the availability of MIMO (Multiple-Input Multiple-Output) systems will further enhance AirFC, as MIMO allows running the FC forward pass in either direction of the link (Fig. 1). As opposed to this flexibility, a MISO system allows running a forward pass in only one link direction. Due to implementation and experimental validation constraints, the rest of the paper will only consider a MISO system, but the operations can be trivially extended to MIMO as well. The system architecture is summarized in Fig. 6, where we see that the computation of each FC layer in the forward pass is implemented by a set of synchronized transmissions towards a certain receiver. If indeed multiple antenna elements are available on the transmitter and receiver side, each FC layer can be implemented through back and forth transmissions between both agents. Here, the System Orchestrator is assumed to have full knowledge of model parameters and system architecture (i.e., available TX/RX-nodes and radios), and allocates the required resources based on the structure of the neural network. It also orchestrates any re-transmissions if needed, specifically for the case $L > N$ as described in Sec. 5. Consequently, the System Orchestrator distributes the required model weights and FC layer inputs to the corresponding node-TXs. A node

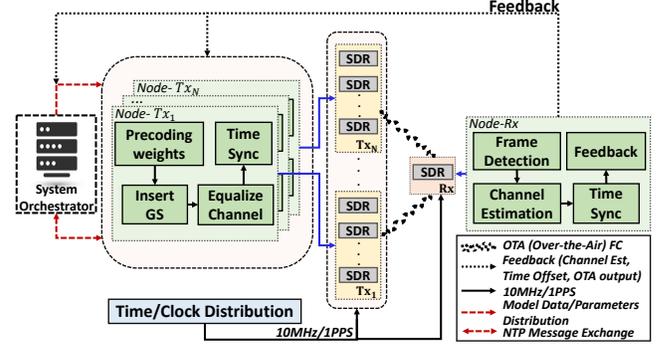


Figure 6: AirFC system architecture showing hardware and software components of the system.

consists of a host machine connected to a number of SDRs with β antennas. Each node-TX performs supporting operations needed to fully implement AirFC (i.e. precoding, channel pre-equalization) before starting the transmissions on each SDR. In turn, for every transmitted stream, the node-RX will perform synchronization operations and channel estimation, which will be part of the feedback to the node-TXs for the following transmission round. As mentioned before, all radios need to be synchronized in time and frequency to ensure coherent detection. This is achieved by connecting all devices to a common clock and time reference, i.e., [18].

• **Channel model and compensation.** As mentioned previously, AirFC employs OFDM waveform. Thus, the wireless channel from transmitter i and subcarrier k can be expressed as a complex number $h_{i,k}$ (Eq. 3). Given that the channel estimates are required at the node-TXs to compute the precoding weights w , the node-RX will send back the channel information as feedback. We define the channel information as $\mathbf{H} = [h_1, h_2, \dots, h_\beta]$, where h_i is a vector of channel estimates for antenna i and all sub-carriers. In order to estimate \mathbf{H} for all transmitters, every antenna element sends a reference signal with good cross-correlation properties (i.e. Gold sequences). This allows the receiver to detect such sequences and isolate each one of them to conduct multiple independent estimates.

4 FEASIBILITY STUDY

In this section, we empirically analyze the different sources of noise considering experimental data and the system model described in Sec. 3. Additionally, we provide initial results to analyze the feasibility of AirFC.

4.1 Error Characterization

As described in Eq. 2, our system model considers two sources of noise, which include the receiver thermal noise (N_z) and the imperfect channel estimation ($\hat{h} = h + h_\epsilon$). In order to evaluate our approach under realistic conditions, we empirically measure such distortions with real data collected using SDRs. For details about the experimental setup and testbed, we refer the reader to Sec. 6.

• **Channel estimation error:** We conduct a data collection campaign of total of 10K channel measurements using least squares (LS) estimation. The data was collected under static conditions, and the channel coherence time was assumed to be greater than the measurement time. Additionally, in order to characterize the effect of using multiple Gold sequences to estimate each independent

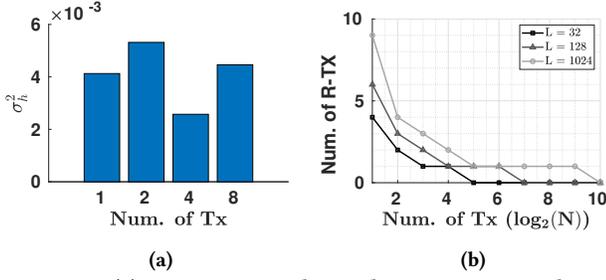


Figure 7: (a) Variance in channel measurement data for the different number of transmitters. (b) Number of re-transmissions (R-TX) needed ($\log_N(L) - 1$) vs number of Tx ($\log_2(N)$) for different sequences length (L).

transmitter to receiver channel, we measure the channel variation while using up to 8 transmitters at a time. We present the results in Fig. 7a. It can be observed that there is no dependency on the number of transmitters used given the strong orthogonality of the used Gold sequences.

• **Receiver thermal noise:** Additionally to the channel estimation error, we also conduct a noise measurement campaign in order to conduct simulation-based studies that will consider realistic noise levels. The additive noise was measured by collecting IQ samples at the receiver, while keeping all transmitters off. Measurement data indicates a noise power of -65dB.

4.2 Preliminary Results

In order to evaluate the feasibility of AirFC, we conduct a simulation study considering the realistic channel and hardware impairments described in the previous subsection. In particular, we train a neural network on MNIST [19], a dataset of 10 different handwritten digits consisting of 28x28 black and white images. MNIST has been widely adopted as a baseline to evaluate different machine learning approaches, and we choose this dataset to highlight the flexibility of AirFC to work on a variety of input types, including images. We train a 2 layer perceptron with an input size of 784 and 16 and 10 neurons on each layer. Additionally, we add (i) Gaussian noise to the neuron weights proportional to the channel estimation error, and (ii) additive noise at every neuron, equivalent to the receiver thermal noise. We obtain an accuracy of 93.51% after 20 epochs. Additionally, we show that a deeper model, with 5 FC layers instead of 2, achieved an accuracy of 98.2% under the same channel and hardware impairments. However, to facilitate implementation efforts (Sec. 6), the remaining of this paper will focus on the two layer network. In closing, we reiterate that our contribution is on reproducing comparable results to a digital model implementation, and not in advancing the state-of-the-art of a specific model itself.

5 FC WEIGHTED SUM FOR $L > N$

In this section, we propose a re-transmission approach for cases when the number of antennas is not sufficient to compute the FC or convolutional operations ($N < L$). We analytically derive how the channel and receiver noises impact the resulting computation.

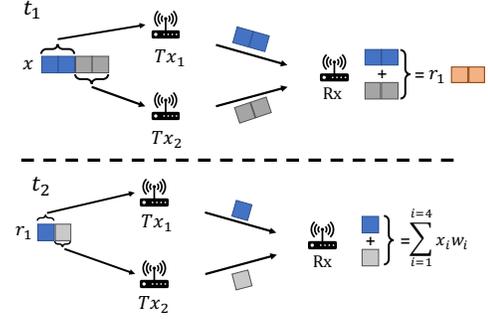


Figure 8: Re-Transmission approach to compute weighted approach when $L > N$. Here, $L = 4$ and $N = 2$. In the first iteration (t_1), Tx_1 transmits x_1 and x_2 , whereas Tx_2 transmits x_3 and x_4 . The two sequences add up together at the Rx, constructing a new sequence (r_1) with length 2. In iteration t_2 , each Tx transmits one sample and the receiver obtains the weights sum of the input sequence x . The pre-equalization steps and the receiver noise are not shown in this diagram.

5.1 Number of Iterations

In Eq. 2, we assumed $N \geq L$. Therefore, the weighted sum is obtained through a single transmission, where every antenna transmits an individual sample from the input sequence x . However, FC layers can be formed by hundreds of neurons, and in the general case, even state-of-the-art mMIMO (massive MIMO) systems may not scale proportionally [20]. In order to resolve this mismatch between L and N , we propose a re-transmission based approach where the input sequence is recurrently truncated into N sub-sequences, until the sub-sequence length is $\leq N$. We describe this in Fig. 8.

Consider a sequence x of length L and a set of N transmitters, with $N < L$. We divide the sequence x in N sub-sequences of length $L_1 = \lceil L/N \rceil$, with each one of them subsequently transmitted by a different radio. Then, the received sequence r of length L_1 contains the element-wise sum of the sub-sequences transmitted by each individual transmitter. Next, r is divided again into L sub-sequences, reducing the received sequence r_2 to $L_2 = \lceil L/N^2 \rceil$ samples. This process can be repeated iteratively, until the length of the sequence r_i is smaller than the number of transmitters ($L_i < N$), which results in a single value on the receiver side. The total number of re-transmissions is expressed as:

$$R = \lceil \log_N(L) \rceil - 1 \quad (4)$$

In Fig. 7b, we analyze how R varies for different L and N values. Note that if $N \geq L$, one single transmission is enough and no re-transmissions are needed ($\lceil \log_N(L) \rceil = 1$), as discussed in Sec. 3.1. While this approach resolves the disparity between the sequence length and the number of transmitters, the final weighted sum is distorted by multiple rounds of channel equalization error and the noise at the receiver. We derive an analytical formulation to analyze such errors in the following subsection.

5.2 Error Analysis

Here, we analyze the approach presented in the previous subsection under perfect and imperfect channel knowledge. We consider the channel model described in Sec. 3.1. In particular, we express the estimated channel \hat{h} as the perfect channel h plus the estimation error h_e (i.e. $\hat{h} = h + h_e$). Then, the received signal r is expressed as:

$$r = \sum_{i=1}^N x_i h_i \frac{(h_i^* + h_\epsilon)}{\|\hat{h}_i\|^2} + \mathcal{N}_z \quad (5)$$

$$= \sum_{i=1}^N \frac{x_i h_i h_i^*}{\|\hat{h}_i\|^2} + \frac{x_i h_i h_\epsilon}{\|\hat{h}_i\|^2} + \mathcal{N}_z, \quad (6)$$

where the second term is a noise factor due to the channel estimation error and \mathcal{N}_z is the receiver noise. After the experimental characterization in Sec. 4.1, we observe that channel error follows a Gaussian distribution:

$$r = \sum_{i=1}^N x_i + \mathcal{N}(0, \sum_{i=1}^N x_i^2 \sigma_h^2 + \sigma_z^2), \quad (7)$$

where the first term is the desired signal, the second term represents the receiver thermal noise and the beamforming noise due to the channel estimation error. After the first re-transmission (r_2), the received signal can be expressed as:

$$r_2 = \sum_{i=0}^N r^{i/N} h_i \left(\frac{h_i^*}{\|\hat{h}_i\|^2} + \frac{h_\epsilon}{\|\hat{h}_i\|^2} \right) + \mathcal{N}_z, \quad (8)$$

where $r^{i/N}$ represents the i th sub-sequence of r . From Eq. 7-8, we observe that no close-form formulation is possible for the accumulated noise in terms of the number of re-transmissions. Hence, we evaluate numerically the impact of such re-transmissions in Sec. 7. Next, we give a close-form formulation of the total accumulated noise per number of re-transmissions under the assumption of perfect channel knowledge ($\sigma_h^2 = 0$). The received signal can be expressed as $r = \sum_{i=1}^N x_i + \mathcal{N}(0, \sigma_z^2)$. Then, r_2 becomes:

$$r_2 = \sum_{i=1}^N r_1^{i/N} + \mathcal{N}_z = \sum_{i=1}^N x_1^{i/N} + (N+1) * \mathcal{N}_{z_1} \quad (9)$$

where \mathcal{N}_{z_1} is a Gaussian distribution with the same statistics as \mathcal{N}_z . Following this re-transmission pattern, the resulting weighted signal sum y_t after all the required re-transmissions can be expressed as:

$$y_t = \sum_{i=1}^L x_i + \mathcal{N}_{z_t} \quad (10)$$

$$\sigma_{z_t}^2 = \sigma_z^2 \sum_{k=0}^R N^k = \sigma_z^2 \left(\frac{1 - N^{R+1}}{1 - N} \right), \quad (11)$$

where $\mathcal{N}_{z_t} = \mathcal{N}(0, \sigma_{z_t}^2)$ is the total accumulated noise that follows a Gaussian distribution with variance $\sigma_{z_t}^2$, which can be computed following a geometric series. In conclusion, that after R re-transmissions, the obtained weighted sum will include additive noise with mean zero and standard deviation $\sigma_{z_t}^2$.

6 SYSTEM IMPLEMENTATION

6.1 AirFC Validation on the ORBIT Testbed

In this section, we describe the AirFC implementation details using COTS SDRs in the ORBIT (Open-Access Research Testbed for Next-Generation Wireless Networks) testbed [21].

•**Large-scale testbed:** ORBIT consists of 400 radio nodes placed in a 20×20 grid with 1m spacing between the adjacent nodes. These nodes are equipped with various radio platforms including 802.11 a/b/g, Bluetooth, Zigbee, and various versions of SDRs, i.e., USRPs. For the implementation presented in this work, we utilize a total of four mini-racks of SDRs, where each of the latter contains eight

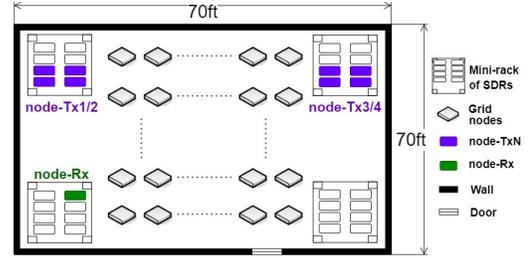


Figure 9: Outline of ORBIT grid testbed utilized for AirFC implementation. There are four mini-racks located in the corners of the grid. Each rack contains 8 Ettus USRP X310 SDRs attached with two UBX160 radio daughterboards. Figure showcases the allocation of AirFC nodes used to realize complex-valued neural network shown in Fig. 11.

Ettus USRP X310 SDRs [22]. All resources in ORBIT are accessed remotely via SSH. All nodes are configured using the experiment controller service (OMF), which allows us to configure radios with desired settings.

•**Tx-Rx hardware as neurons:** We select a total of nine radios from the mini-racks, where eight radios are transmitters and the ninth one is the receiver, as illustrated in Fig. 9. Given that each X310 radio has two RF chains with UBX 160MHz daughterboards, we use a total of 16 transmitting antennas. All USRPs in a mini-rack are connected to the ORBIT network via 10Gbps Ethernet connections, which grants access to them from any selected node. We select four X310 radios from two different locations that provide moderate SNR conditions. Each cluster of four radios is controlled by a single host machine. Thus, we use a total of 8 radios with 16 transmitting antennas, with each one of them transmitting signals pertaining to one FC layer neuron, followed by reception at one receiver, which gives the cumulative output of the neurons.

•**Synchronized SDRs:** A mini-rack has its local reference clock (Octoclock-G CDA-2990 [23]) that provides PPS and 10MHz clock synchronization signals to all its installed SDRs. In turn, all Octoclocks are connected to a master reference clock, which provides common reference signals to each one of them. The use of equal length cables and a symmetric topology ensures that all connected devices will see the same reference signals with very little deviation in phase and time.

6.2 AirFC Framework

Here, we describe the framework required to implement AirFC.

•**System orchestration:** All network connected nodes in ORBIT domain can communicate through Ethernet connections. This allows us to: (i) orchestrate data distribution among the transmitter nodes, (ii) run coordinated transmissions, and (iii) relay feedback messages from receiver to transmitters. The neural network model is known at the orchestrator, which runs in one of the host machines connected to the radios. Then, it sends the desired IQ symbols to each transmitter considering the number of neurons, layers, input data size and number of transmitters. To achieve synchronized start time of an AirFC transmission (beside hardware-level synchronization achieved using Octoclocks), we establish software-based time synchronization by combining Network Time Protocol (NTP) and Precision Time Protocol (PTP), providing hardware time-stamping for all nodes.

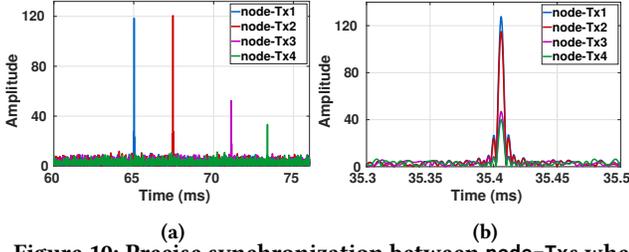


Figure 10: Precise synchronization between node-Txs where each node-Tx consists of 4 transmit antennas: (a) Correlation inaccuracy due to sample-level misalignment, (b) Aligned cross-correlation peaks after time correction.

- Frame structure:** Each transmitted frame consists of a training sequence, acting as a preamble, and an appended payload of IQ symbols, which are the input values of each neuron. The symbols are OFDM modulated. Each block has a cyclic prefix (CP) at the OFDM modulation to mitigate inter symbol interference (ISI). Further, we use pre-defined Gold sequences [24] as preambles for synchronization and channel estimation. Each of these unique sequences is assigned to the payload of IQ symbols at each transmitter antenna. Due to their well-known autocorrelation and crosscorrelation properties, Gold sequences allow to estimate channels accurately and detect each transmitted stream concurrently, which will later enable synchronization between each data stream.

- Channel estimation and feedback:** Channel estimation is conducted at the node-RX (output neuron). Cross-correlation between the received preambles and pre-defined reference Gold sequences is conducted to obtain a specific signal transmitted on a given antenna. Here, we use Least Squares (LS) to obtain the multiple independent channel estimates from each antenna i to the receiver node i.e. $\hat{H} = \hat{G}/G$, where \hat{H} is the estimated response, G the transmitted Gold Sequences and $\hat{G} = [\hat{g}_1, \hat{g}_2, \dots, \hat{g}_\beta]$ are the received training symbols. In our implementation of AirFC, we assume the bandwidth of the transmitted signal to be smaller than the coherence bandwidth. Thus, the channel response from a transmitter antenna to the receiver is assumed to be flat. However, AirFC is independent of the channel estimation technique and is broadly applicable independent of the channel conditions or bandwidth of the signal used. Finally, in order to ensure coherent detection, the node-TX estimates the misalignment between the received frames by inspecting the correlation indices in the time domain. Then, the receiver sends back a feedback packet containing (i) the channel estimation that is estimated from channel response of each transmitter, (ii) a time correction for preventing misalignment between preambles, and (iii) if re-transmission is needed (see Sec. 5), sum of sub-sequence transmitted by each individual transmitter at the receiver as an input to next transmission.

- Timing and synchronization:** Recall, all radios share a common clock which distributes 10MHz and 1PPS signal externally to enable hardware-level frequency and time synchronization among them. In addition to hardware-level synchronization, we provide software-level synchronization among transmitters to enable synchronized transmission time. Software-level synchronization is necessary given that we require sample-level precision, which translated into μs -level accuracy (see Fig. 10a). This is achieved by inspecting cross-correlation indices for all received Gold sequences

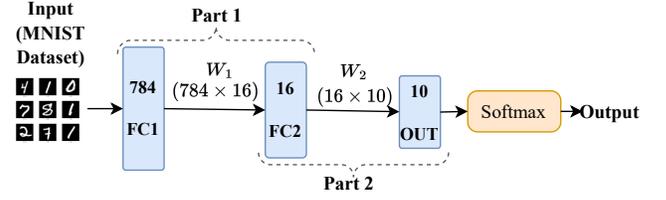


Figure 11: The neural network architecture model with complex weights used for image recognition. Each the FC and OUt layers are complex-valued FC layers. The input is converted to complex domain to be fed to the complex-valued FC layers. We represent the two weight matrices of the network as Part1 and Part2 to design the validation pipeline for performance analysis.

and identifying the sample-level misalignment among all received streams. Such time corrections are sent back to the transmitters, which will zero-pad their respective sequences to correct for the detected misalignment. As shown in Fig. 10b, we achieve sample-level synchronization after correcting for the detected sequence misalignment.

6.3 AirFC for Image Recognition

Next, we demonstrate how the over-the-air weighted sum in AirFC is accurate enough to replace its digital equivalent for a real-world problem of image recognition.

- Dataset description and model architecture:** We consider the well known MNIST dataset [25] of handwritten digits that contains training and test set of 60,000 and 10,000 examples, respectively. We use a two-layer complex-valued neural network architecture [26] where the two weight matrices are realized by over-the-air computation of proposed AirFC framework. As the overall goal of AirFC is to realize the weighted sum computation of different weight matrices in the subsequent layers, we choose a 2-layer architecture for easier understanding. The model architecture of the NN architecture is presented in Fig. 11, where the complex-valued weight matrices W_1 and W_2 are replaced with over-the-air transmission of AirFC while validation. To make the model robust to the changing wireless environment, we add noise to the weights of the layer as well as additive noise while training. The Pytorch implementation of our training and validation pipeline is available to the research community [27].

- Generation of trained model:** We create a trained model on the MNIST dataset with stochastic gradient descent optimization and *negative log likelihood* loss with 0.001 learning rate. This enables us to obtain log-probabilities by adding a *LogSoftmax* layer in the last layer. The model is trained for 20 epochs, achieving a training accuracy of 96%.

6.3.1 Complex-valued neural networks. Wireless signals are often represented using complex numbers, to express amplitude and phase information. We leverage this property to design a neural network based on complex-valued data and weights, similar to [28], which has the potential to provide superior performance to real valued neural networks [17]. Considering the distributive property of the product operation, the output of a complex-valued fully connected layer can be expressed as:

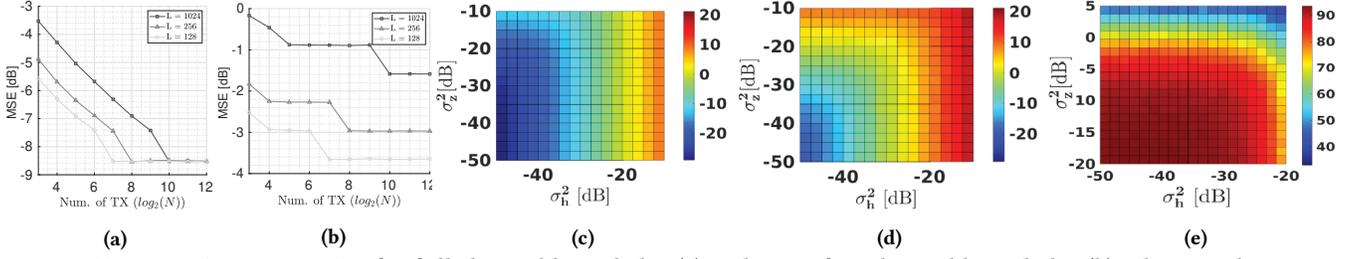


Figure 12: Linear Operation MSE for full channel knowledge (a) and imperfect channel knowledge (b). This simulation uses $\sigma_z^2 = 1e^{-4}$ for both scenarios and $\sigma_h^2 = 1e^{-4}$ for the imperfect channel knowledge case. Linear Operation MSE [dB] under different receiver noise and channel error levels with (c) $N = 2$ and (d) $N = 128$. L was kept constant to 128 in this simulation. (e) shows AirFC performance for different σ_z^2 and σ_h^2 values.

$$y_{FC}^o = \sum_{i=1}^N x_R \times w_R^{o,i} - x_I \times w_I^{o,i} + j(x_R \times w_I^{o,i} + x_I \times w_R^{o,i}), \quad (12)$$

where y_{FC}^o is the output of neuron o , $w^{o,i}$ is the weight of the edge from input i to output o and N the number of input neurons. We leverage the expression in Eq. 12, where the real and imaginary values are treated independently, to implement complex-valued layers using two real-valued layers, where each one of them represents the real (ϕ_{w_R}) and imaginary parts (ϕ_{w_I}). We refer the reader to the work in [28] for further details.

6.3.2 Weight clipping. Here, we discuss how the power limitations of realistic testbeds constrain the weights that can be implemented in AirFC. First, we describe the transmitted power (P_{Tx}) as:

$$P_{Tx} = |p|^2 |x|^2 = \underbrace{|w|^2}_{P_w} \times \underbrace{\left| \frac{\hat{h}}{|h|^2} \right|^2}_{P_{eq}} \times \underbrace{|x|^2}_{P_x}, \quad (13)$$

where P_w represents the power associated with the FC layer weights, P_{eq} is the power of the pre-equalization term and the P_x represents the power of x . We see that w is the only term added by AirFC on top of any regular communication operation. Then, given a certain power budget limited by γ_{max} , the transmitted power should always remain lower than the designated power constrain ($P_{Tx} \leq \gamma_{max}$). Thus, w is constrained to: $|w| \leq \sqrt{\frac{\gamma_{max}}{P_{eq} P_x}}$.

In this paper, we do not include any power requirement threshold for AirFC. We achieve that by imposing a constrain on the weights of the neural network, such that they stay within the unitary value circle ($|w| \leq 1$). This constraint is imposed during the model training phase by clipping the weights that go beyond the designated threshold.

7 PERFORMANCE EVALUATION

This section evaluates the equivalence between the operations computed by AirFC and its digital counterpart.

7.1 Validating Accuracy of Linear Operation

7.1.1 Number of transmitters. We evaluate the performance of an independent weighted sum, considering different number of transmitters are available, as well as different sequence lengths. Fig. 12a-12b show the mean square error (MSE) under full and imperfect channel knowledge for varying N . It can be observed that when only the receiver noise is considered, the accumulated error decreases as N increases, as expected, until $N \geq L$. On the contrary, when the channel estimation error is taken into consideration,

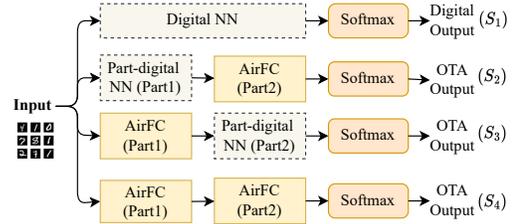


Figure 13: The validation pipeline of AirFC.

the MSE only reduces noticeably when the total number of re-transmissions decreases, as described in Eq. 4.

7.1.2 Noise level. The different noise levels impact the received signal generating errors when compared to the clean output expected on the received side. In Fig. 12c-e, we evaluate the impact of the receiver noise (σ_z^2) and the channel estimation error (σ_h^2) on the weighted sum at the receiver side. It can be observed that for the noise levels measured in Sec. 4.1, the MSE does not exceed ≈ -10 dB for neither $N = 2$, nor $N = 128$. Given that the tolerable error uniquely depends on the targeted application, in the following subsection we evaluate how the different noises affect the performance of a neural network.

7.2 Simulation Results

We evaluate the performance of the model in Fig. 11 under different noise conditions. In particular, we test the model with different additive noise and beamforming noise in simulation. The model performance is tested for additive noise from -20 dB to 5 dB and the channel estimation error from -50 dB to -20 dB. The noise ranges were selected after identifying where the trained model performance is heavily impacted. It can be observed that for $\sigma_h^2 = -31$ dB, an accuracy $> 90\%$ is achieved for $\sigma_z^2 < -9.5$ dB. Equivalently, for $\sigma_z^2 = -9.5$ dB, an accuracy $> 90\%$ is achieved for $\sigma_h^2 < -31$ dB.

7.3 OTA Validation Pipeline and Results

The trained model (discussed in Sec. 6.3) is deployed over the real-world large testbed (discussed in Sec. 6.1) for inference following the AirFC framework. We use 10,000 examples of the test set of MNIST dataset for validating the proposed AirFC framework. The validation pipeline is shown in Fig. 13. The first setting (S_1) of our validation pipeline infers how the trained model performs in all-digital environment. In the next two settings (S_2 and S_3), we realize either of weight matrices W_1 or W_2 through OTA transmission within the AirFC framework at the hidden or output layers,

Settings	Validation Pipeline	Accuracy (%)
S_1	Digital NN	93.34
S_2	Digital (Part1) + AirFC (Part 2)	91.29
S_3	AirFC (Part1) + Digital (Part 2)	92.83
S_4	AirFC (Part1) + AirFC (Part 2)	92.61

Table 1: The accuracies of different settings following the validation pipeline in Fig. 13.

respectively. In the final setting (S_4), we implement AirFC OTA transmission for both the W_1 and W_2 and perform the weighted sum at the output layer. This S_4 setting can also be referred as *AirFC end-to-end* implementation, where the end-to-end neural network model is deployed in a real-world OTA environment. The results of those different settings are shown in Tab. 1.

Overall, from the different settings of the proposed validation pipeline, we observe AirFC returns competitive test accuracy with only a 0.73 reduction when compared with the digital counterpart, which is negligible for small to medium-scale implementation..

7.4 Discussions

The advantage of AirFC over the conventional computing platforms is that AirFC can perform computation of a FC layer in one single transmission that generally requires many consecutive operations on a digital platform. In this work, we do not optimize the system design in terms of energy and inference time; rather, we provide a proof-of-concept of AirFC with the real-world implementation and leave the enhanced engineering design and optimization for future work. We next analyze the system complexity of AirFC and discuss how AirFC can potentially compete with digital computing platforms by leveraging upcoming advanced wireless technologies.

•**Complexity analysis:** We define the complexity of a classifier *during inference* by the number of computations required to classify input data. For example, a $n \times n$ matrix multiplication of one FC layer requires $O(n^2)$ digital MAC (multiply-accumulate) operations, where a dot product between the input layer and weight matrix requires n multiplications and $n - 1$ additions. Since they are n elements, the dot product needs to be computed n times, resulting in $n(2n - 1)$. Today, one MAC operation is performed in one clock cycle of the processor, e.g., a computer having clock rate in the order of 1GHz can perform 10^9 MAC/s. In contrast, AirFC's computation time is bound to a digital base band signal with available bandwidth, which is directly related to sampling rate of the radio, and OFDM block design. To expand this, we express subcarrier frequency spacing, $\Delta F = B/N_{FFT}$, where B is bandwidth utilized to perform OTA inference and N_{FFT} as the size of Fast Fourier Transform (FFT). Then, we derive the OTA inference time of one FC layer as follows:

$$\tau = \begin{cases} \frac{1}{\Delta F}, & \text{if } N \geq L \\ \sum_{k=1}^R \frac{1}{\Delta F} L_k, & \text{if } N < L, \end{cases} \quad (14)$$

where L_k and R represent the length of sequences for k th R-TX and total number of re-transmissions, respectively. Note that we assume $M \geq N_{FFT}$ for a given FC layer with N input neurons and M output neurons. Considering the parameters of the implemented system in this work where bandwidth, B , is 1MHz and FFT size, N_{FFT} , is 128, we calculate the inference time of MNIST data set with proposed complex-valued NN architecture as 6.8ms for the Part1 and 128 μ s for the Part2 (see Fig. 11). The number of R-TX is directly proportional to increased inference time from Part 1.

This can be reduced by increasing number of antennas allocated in network, thereby reducing number of R-TX. Furthermore, the computation time of Part1 and Part2 can reduce to 169.6 μ s and 3.2 μ s, respectively, for WiFi standard 802.11ac/x [29].

Another system parameter we need to consider is the computation for *channel estimation*. In AirFC framework, our implementation requires two consecutive computations: (i) correlation of training sequences, and (ii) LS estimation for channel as explained in sec 6.2. Algorithmic complexity for both operations is $O(K)$ where K represents the length of training sequences. We note that one estimated channel value per transmitter can be used to equalize many transmissions as long as the channel is flat and the condition, $t_{corr} < T_c$, holds that computation time of correlation should be within channel coherence time. Therefore, the computation time of channel estimation should not be considered only for one OTA inference, but numerous AirFC transmissions. Additionally, time-efficient channel estimation techniques, such as channel sounding or channel reciprocity [30], can be implemented for AirFC to reduce computation time, instead of feedback-reliant approaches.

•**Future directions:** Emerging technologies like THz communication is making rapid strides in terms of device technology [31] and channel modeling and propagation [32, 33]. The authors in [34] reported ultrabroadband communication above 100GHz with different channel bandwidths (from 2GHz to 32GHz). When we apply the proposed physical layer parameters from [34], i.e., 32MHz subcarrier spacing, 64-point FFT size and sampling clock at 2.048GHz, the computation time of Part1 and Part2 reduce to 1.7 μ s and 31.3ns, respectively. The digital counterpart running with a clock rate of 1GHz consumes 12.7 μ s. Thus, for larger NNs, AirFC will benefit from the OTA inference to compute $n \times n$ FC layer in nano-second duration while the computation time of same network in an all-digital platform will increase as the NN becomes larger.

8 RELATED WORKS

A number of works have proposed OTA computations (AirComp) using RF signals for numerous applications [35][36][37][38][39]. The work in [35] enables low-latency uploads and aggregating the weights of the machine learning models by exploiting simultaneous co-channel transmission and waveform superposition. In AirComp [36], the authors perform wireless data aggregation using a MIMO architecture, whereas the work in [37] leverages the AirComp concept for averaging the model weights in a federated learning setup. The authors in [38] describe OTA implementation of a set of linear and non-linear functions, including variance, linear regression and max function. Similarly, the work in [39] presents a beamforming framework for multi-function OTA computation and applies it to MIMO multi-modal sensor networks, highlighting the relevance of such architectures for supporting distributed learning and inference. However, all aforementioned works are limited to analytical studies, only validated in simulation. To the best of our knowledge, in this paper we present the first experimental validation of OTA computation and apply it for inference tasks using a NN.

Experimental approaches that perform training or inference on NN have been proposed in [40][41][42][17]. In [40], authors use stochastic gradient descent to train an analog NN by varying the conductance of programmable resistive diodes. The work in [41] demonstrates an analog recurrent NN using acoustic signals by

leveraging wave physics properties, whereas [42] presents a RIS-based experimental prototype testbed to perform OTA convolutions, which raises concerns of scalability. In [17], the authors implement an optical neural chip to realise complex-valued NN. However, they do not address the limitation of large FC layers, and thus, their solution limits the dimensions of the FC layers. Motivated by the shortcomings in the state-of-the-art, we present AirFC in this paper, which is the first-of-its-kind experimental and scalable implementation of over-the-air FC layers.

9 CONCLUSIONS

In this paper, we demonstrate the first-of-its-kind over-the-air neural network inference using FC layers on a real system using COTS devices. In particular, we leverage a programmable 16-transmitter MISO system and we show the feasibility of engineering transmissions to accurately replicate the linear operations of a FC. Furthermore, we propose and characterize a re-transmission based approach that enables FC layer computation when the number of available antennas is smaller than the input dimensions. Overall, AirFC achieves an experimentally observed accuracy of 92.61% on the MNIST dataset, which represents only a 0.73% accuracy loss compared to its digital counterpart.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the funding from the US National Science Foundation (grant CNS-2112471). We are also grateful to Joshua Groen for his great help and insightful suggestions.

REFERENCES

- [1] Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals," in *Proceedings of the 19th Annual International Conference on Mobile Computing Networking*, ser. MobiCom '13, NY, USA, 2013, p. 27–38.
- [2] H. X. Nguyen, R. Trestian, D. To, and M. Tatipamula, "Digital twin for 5g and beyond," *IEEE Communications Magazine*, vol. 59, no. 2, pp. 10–15, 2021.
- [3] X. Shuai, Y. Shen, Y. Tang, S. Shi, L. Ji, and G. Xing, "Milliye: A lightweight mmwave radar and camera fusion system for robust object detection," ser. IoTDI '21, NY, USA, 2021, p. 145–157.
- [4] T. Wild, V. Braun, and H. Viswanathan, "Joint design of communication and sensing for beyond 5g and 6g systems," *IEEE Access*, pp. 30 845–30 857, 2021.
- [5] C. Chaccour, W. Saad, O. Semiari, M. Bennis, and P. Popovski, "Joint sensing and communication for situational awareness in wireless thz systems," in *ICC 2022 - IEEE International Conference on Communications*, Seoul, South Korea, 2022.
- [6] M. Goldenbaum, H. Boche, and S. Stańczak, "Harnessing interference for analog function computation in wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 61, no. 20, pp. 4893–4906, 2013.
- [7] G. Zhu, J. Xu, K. Huang, and S. Cui, "Over-the-air computing for wireless data aggregation in massive iot," *IEEE Wireless Communications*, pp. 57–65, 2021.
- [8] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *Trans. Wireless Comm.*, vol. 19, no. 3, p. 2022–2035, mar 2020.
- [9] I. F. Akyildiz and J. M. Jornet, "Realizing ultra-massive mimo (1024*1024) communication in the (0.06-10) terahertz band," *Nano Commun. Networks*, 2016.
- [10] S. Abadal, E. Alarcón, A. Cabellos-Aparicio, M. C. Lemme, and M. Nemirovsky, "Graphene-enabled wireless communication for massive multicore architectures," *IEEE Communications Magazine*, vol. 51, no. 11, pp. 137–143, 2013.
- [11] S. Deb, K. Chang, X. Yu, S. P. Sah, M. Cosic, A. Ganguly, P. P. Pande, B. Belzer, and D. Heo, "Design of an energy-efficient cmos-compatible noc architecture with millimeter-wave wireless interconnects," *IEEE Transactions on Computers*, vol. 62, no. 12, pp. 2382–2396, 2013.
- [12] L. Ju, B. Geng, J. S. A. Horng, Çağlar Girit, M. C. Martin, Z. Hao, H. A. Bechtel, X. Liang, A. Zettl, Y. R. Shen, and F. Wang, "Graphene plasmonics for tunable terahertz metamaterials," *Nature nanotechnology*, vol. 6 10, pp. 630–4, 2011.
- [13] ETSI, "5g; study on new radio (nr) access technology (3gpp tr 38.912 version 14.0.0 release 14)," May 2017. [Online]. Available: https://www.etsi.org/deliver/etsi_tr/138900_138999/138912/14.00.00_60/tr_138912v140000p.pdf
- [14] F. Mehmeti and C. Rosenberg, "How expensive is consistency? performance analysis of consistent rate provisioning to mobile users in cellular networks," *IEEE Transactions on Mobile Computing*, vol. 18, no. 5, pp. 1098–1115, 2018.
- [15] L. Bonati, P. Johari, M. Polese, S. D'Oro, S. Mohanti, M. Tehrani-Moayyed, D. Villa, S. Shrivastava, C. Tassie, K. Yoder et al., "Colosseum: Large-scale wireless experimentation through hardware-in-the-loop network emulation," in *2021 IEEE International Symposium on DySPAN*, 2021, pp. 105–113.
- [16] srsRAN Project, "Open-source 4g/5g software-radio suites." [Online]. Available: <https://www.srsran.com/>
- [17] H. Zhang, M. Gu, X. Jiang, J. Thompson, H. Cai, S. Paesani, R. Santagati, A. Laing, Y. Zhang, M. Yung et al., "An optical neural chip for implementing complex-valued neural network," *Nature Communications*, vol. 12, no. 1, pp. 1–11, 2021.
- [18] K. Alemdar, D. Varshney, S. Mohanti, U. Muncuk, and K. Chowdhury, "Rflock: Timing, phase and frequency synchronization for distributed wireless networks," ser. MobiCom '21, 2021, p. 15–27.
- [19] L. Deng, "The mnist database of handwritten digit images for machine learning research," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141–142, 2012.
- [20] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive mimo: Benefits and challenges," *IEEE journal of selected topics in signal processing*, vol. 8, no. 5, pp. 742–758, 2014.
- [21] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kremo, R. Sircusa, H. Liu, and M. Singh, "Overview of the orbit radio grid testbed for evaluation of next-generation wireless network protocols," *IEEE Wireless Communications and Networking Conference*, vol. 3, p. 1664–1669, 2005.
- [22] E. Research, "Usrp x310," <https://www.ettus.com/all-products/x310-kit/>.
- [23] —, "Octoclock-g cda-2990," <https://www.ettus.com/all-products/octoclock-g/>.
- [24] Z. Xinyu, "Analysis of m-sequence and gold-sequence in cdma system," *ICCSN*, no. 1, p. 466–468, 2011.
- [25] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, 1998.
- [26] M. W. Matthès, Y. Bromberg, J. de Rosny, and S. M. Popoff, "Learning and avoiding disorder in multimode fibers," *Phys. Rev. X*, vol. 11, p. 021060, Jun 2021.
- [27] D. Roy and G. Reus-Muns, "Implementation of Neural Network Training for AirFC project," https://github.com/debashriroy/AirFC_MNIST, 2022.
- [28] C. Trabelsi, O. Bilaniuk, Y. Zhang, D. Serdyuk, S. Subramanian, J. F. Santos, S. Mehri, N. Rostamzadeh, Y. Bengio, and C. J. Pal, "Deep complex networks," in *International Conference on Learning Representations*, 2018.
- [29] "Ieee standard for information technology—telecommunications and information exchange between systems local and metropolitan area networks—specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 1: Enhancements for high-efficiency wlan," *IEEE Std 802.11ax-2021 (Amendment to IEEE Std 802.11-2020)*, pp. 1–767, 2021.
- [30] D. Vasishth, S. Kumar, H. Rahul, and D. Katabi, "Eliminating channel feedback in next-generation cellular networks," in *Proceedings of the 2016 ACM SIGCOMM Conference*, ser. SIGCOMM '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 398–411.
- [31] A. Singh, M. Andreollo, N. Thawdar, and J. M. Jornet, "Design and operation of a graphene-based plasmonic nano-antenna array for communication in the terahertz band," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 9, pp. 2104–2117, 2020.
- [32] C. Han and Y. Chen, "Propagation modeling for wireless communications in the terahertz band," *IEEE Communications Magazine*, vol. 56, no. 6, pp. 96–101, 2018.
- [33] Y. Wu, J. Kokkonen, C. Han, and M. Juntti, "Interference and coverage analysis for terahertz networks with indoor blockage effects and line-of-sight access point association," *IEEE TWC*, vol. 20, no. 3, pp. 1472–1486, 2021.
- [34] P. Sen, V. Ariyaratna, A. Madanayake, and J. M. Jornet, "A versatile experimental testbed for ultrabroadband communication networks above 100 ghz," *Computer Networks*, vol. 193, p. 108092, 2021.
- [35] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, "Differentially private air-comp federated learning with power adaptation harnessing receiver noise," in *GLOBECOM 2020-2020 IEEE Global Communications Conference*, 2020, pp. 1–6.
- [36] D. Wen, G. Zhu, and K. Huang, "Reduced-dimension design of mimo aircomp for data aggregation in clustered iot networks," in *IEEE GLOBECOM*, 2019, pp. 1–6.
- [37] H. Hellström, V. Fodor, and C. Fischione, "Over-the-air federated learning with retransmissions," in *IEEE SPAWC*, 2021, pp. 291–295.
- [38] O. Abari, H. Rahul, and D. Katabi, "Over-the-air function computation in sensor networks," *arXiv preprint arXiv:1612.02307*, 2016.
- [39] G. Zhu, L. Chen, and K. Huang, "Mimo over-the-air computation: Beamforming optimization on the grassmann manifold," in *IEEE GLOBECOM*, 2018, pp. 1–6.
- [40] J. Kendall, R. Pantone, K. Manickavasagam, Y. Bengio, and B. Scellier, "Training end-to-end analog neural networks with equilibrium propagation," *arXiv*, 2020.
- [41] T. W. Hughes, I. A. Williamson, M. Minkov, and S. Fan, "Wave physics as an analog recurrent neural network," *Science advances*, vol. 5, no. 12, p. eaay6946, 2019.
- [42] S. G. Sanchez, G. Reus-Muns, C. Bocanegra, Y. Li, U. Muncuk, Y. Naderi, Y. Wang, S. Ioannidis, and K. R. Chowdhury, "Airnm: Over-the-air computation for neural networks via reconfigurable intelligent surfaces," *IEEE/ACM Transactions on Networking*, pp. 1–0, 2022.