Received 31 January 2024; revised 16 March 2024; accepted 12 April 2024. Date of publication 22 April 2024; date of current version 14 May 2024.

Digital Object Identifier 10.1109/OJCOMS.2024.3392508

Universal Function Approximation Through Over-the-Air Computing: A Deep Learning Approach

PAVLOS S. BOUZINIS[1,2], NIKOS G. EVGENIDIS[1] (Graduate Student Member, IEEE), NIKOS A. MITSIOU[1] (Graduate Student Member, IEEE), SOTIRIS A. TEGOS[1] (Senior Member, IEEE), PANAGIOTIS D. DIAMANTOULAKIS[1] (Senior Member, IEEE), AND GEORGE K. KARAGIANNIDIS[1,3] (Fellow, IEEE)

[1]Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece

[2]MetaMind Innovations P.C., 50100 Kozani, Greece

[3]Artificial Intelligence and Cyber Systems Research Center, Lebanese American University, Beirut 03797751, Lebanon

CORRESPONDING AUTHOR: N. A. MITSIOU (e-mail: nmitsiou@auth.gr)

ABSTRACT Over-the-air (OTA) computing has emerged as a promising technique that utilizes the superposition property of the wireless multiple access channel (MAC) as a means for computation. In this work, we propose a deep learning-based mechanism that approximates the pre- and post-processing functions of OTA computing, with the ultimate goal of approximating any desired target function. Specifically, we adopt a centralized training-decentralized execution approach that allows independent execution of deep neural networks (DNNs) on both devices and server to interpret the pre- and post-processing functions. The analysis is extended to the case of representing the pre- and post-processing functions to a higher dimensional space, further facilitating the reconstruction of the target function. To evaluate the effectiveness of the proposed method, we introduce a benchmark that serves as a lower bound on the computational distortion, i.e., the average mean square error (MSE) between the target function and the OTA computing estimation, which is described by closed-form solutions. It is noteworthy that the considered benchmark can serve as a reference point for any OTA computing-based application with any target function. Furthermore, the performance of the proposed decentralized DNN over-the-air computing execution (DOTACE) is evaluated through simulations, demonstrating its potential.

INDEX TERMS Over-the-air computing, deep learning, decentralized model execution.

I. INTRODUCTION

DUE TO the rapid and exponential growth in the number of interconnected wireless Internet of Things (IoT) devices, there is a critical need to address and mitigate the challenges posed by communication bottlenecks. At the same time, harnessing and processing massive amounts of wireless data requires the implementation of effective computing solutions. To address these issues, over-the-air (OTA) computing is a promising technique that fuses communication and computation by exploiting the inherent broadcast properties of the wireless medium [1]. More specifically, OTA computing is useful in scenarios where the primary focus is on observing a function derived from the messages of the devices, e.g., the sum of the messages, thus eliminating the need to acquire each individual message. The main mechanism that allows the computation of a target function is the superposition property of the wireless multiple access channel (MAC). In more detail, OTA computing exploits the superposition principle along with appropriate signal processing at both ends in order to reduce the amount of computations required for a desired target function. Except for that, OTA also achieves better resources management due to simultaneous transmission at the same frequency from all devices compared to traditional computing schemes that follow the receive-then-compute paradigm, where data from all devices in the network must be received in serial fashion from every device for the required target function computation to be performed. Remarkably, any real-valued

function can be expressed as a function of a finite sum of univariate functions and, thus, can be approximated by OTA computing [2], [3]. As an example, in real-world deployments involving sensor networks for environmental monitoring or smart city applications, the ability to compute aggregated data (such as average temperature, humidity, or air quality indices) directly over the air can significantly accelerate response times to critical events and reduce the network's energy footprint. Similarly, in the context of federated learning, where preserving privacy while enabling collaborative machine learning models is critical, OTA computing can serve as a key technique to efficiently aggregate model updates without the need to access individual data points, thus preserving user privacy.

A. RELATED WORKS

OTA computing has been examined by the research community from a variety of perspectives, due to its inherent ability to fuse communication and computation and ultimately reduce latency. For instance, in [4], the computational distortion of OTA computing was minimized by optimizing the pre-processing and post-processing functions of the transmitter and receiver, respectively. Moreover, the ergodic performance and the scaling with respect to the number of devices were investigated. In addition, [5] proposed an OTA computing system with multiple antenna arrays at both the transmitter and the receiver, while minimizing the minimum square error (MSE) between the target function and the estimation. Also, OTA computing has been examined in multiple-input multiple-output (MIMO) systems for a low-complexity and low-latency computation of a vector-valued function [6]. Additionally, OTA computing has been proposed to facilitate federated learning (FL) applications by transmitting of the local training parameters over-the-air [7], [8], [9], [10], [11], [12], [13], [14]. Specifically, the computational distortion on the convergence rate of over-the-air-aided FL has been studied in [7], [8], [9], [10], while in [11], [12], [13], [14] various aspects of the over-the-air-aided FL systems have been investigated, such as the users' heterogeneity, and the joint optimization of the communication and the FL procedure. Additionally, OTA computing has been investigated as a means to reduce the completion time of solving distributed resource allocation problems in wireless networks by employing distributed optimization techniques [15], [16]. Moreover, OTA computing was utilized in [17] to improve the communication efficiency of task-oriented communications.

However, it is highlighted that the vast majority of the existing literature on OTA computing has examined only certain types of functions, such as sum, weighted average, and geometric mean. This limitation hinders the potential and broad applicability of OTA computing, despite its promise to approximate almost any function. In this direction, in [18] an algorithm for the approximation of an arbitrary function through a nomographic representation was proposed. However, its scalability is hindered by the complexity of the proposed algorithm. Furthermore, the algorithm was developed based on an ideal scenario, neglecting the inherent constraints of a wireless medium. Incorporating deep learning-based function approximation seems to be a promising way to address this challenge and fully exploit the capabilities of OTA computing [19]. It is clarified that to the best of our knowledge, only [18] and [19] focused on investigating techniques that approximate arbitrary functions for OTA computing applications.

In recent years, deep neural networks (DNNs) have achieved remarkable success in a wide range of wireless network applications. The work in [20], [21] trained a DNN as an autoencoder to represent and jointly optimize the transmitter and receiver of a communication system in an end-to-end fashion. Furthermore, deep learning has shown potential in facilitating channel estimation and signal detection in OFDM systems [22], while DNNs have exhibited promising capabilities in performing power allocation for interference management [23]. Motivated by the encouraging capabilities of deep learning, [19] proposed the use of DNNs to approximate the pre- and post-processing functions of OTA computing. The effectiveness of this approach was evaluated through simulations in applications such as anomaly detection and distributed regression.

B. MOTIVATION AND CONTRIBUTION

It is apparent that in order to fully harness the potential of OTA computing in a wireless MAC, novel techniques should be employed that can accurately approximate any target function. This will ultimately transform OTA computing into a practical universal function approximator. Reference [19] paved the way in using DNNs for this purpose. However, the architectural and structural details of the considered DNNs, as well as the precise procedures of training and inference, have not been adequately clarified. In addition, the constraints imposed by the wireless medium have not been sufficiently addressed. In particular, the main challenge lies in the fact that each device and the server should possess an individual trained DNN to represent the pre- and post-processing functions, while the training should be performed by a single entity in a centralized manner, to capture the characteristics of the system as a whole. For this reason, attention should be given in training a centralized DNN in a manner that allows the split of its architecture between the users and the server. This process will facilitate the creation of individual DNNs both at the users' and the server's sides, that are pre-trained and can be utilized individually by all entities. The significance of this aspect was not stressed in [19]. In essence, the centralized training should emulate the OTA computing procedure in a wireless environment, learning both the pre- and post-processing functions. Following that, the practical realization of this lies in the decentralized execution of the partitioned DNN among users and the server during inference.

To address those issues, the contributions of this paper are as follows:

- We propose a DNN-aided OTA computing mechanism consisting of two stages, which facilitates the achievement of universal function approximation. First, we design a centralized DNN (cDNN) that is trained by the server. The cDNN is constructed in a way that resembles an OTA computing implementation in a wireless MAC. Second, upon completion of training, parts of the cDNN along with their respective trained weights are partitioned and distributed to the users and the server, representing the pre- and post-processing functions of OTA computing, respectively. This process enables decentralized DNN OTA computing execution (DOTACE). The final output of the DOTACE is the server's estimation of the target function. Significantly, the tailored design of the above mechanism ensures that the DOTACE performs identically to the cDNN in terms of average MSE between the estimated and the target function. Finally, the scope of the analysis is broadened to include the representation of pre- and post-processing functions within a higher dimensional space, which we call *augmented vector* representation. This expansion aims at facilitating a more precise reconstruction of the target function by introducing redundancy.
- To evaluate the performance of the DOTACE, we design a benchmark, termed as oracle benchmark (OB). The latter considers the presence of a "superuser" with knowledge of all devices' messages. As a result, this superuser can accurately compute the target function without any loss and transmit it to the server through analog transmission. Following that, we derive a closed-form solution for the minimum average MSE that this benchmark can achieve. It is clarified that no OTA computing-related implementation can surpass the performance of the OB, making it universally applicable as a performance bound and potentially enabling its widespread use.
- The performance of the DOTACE is evaluated through simulations in terms of average achieved MSE. We select an example of a target function whose representation as a nomographic function is rather intractable, encouraging the utilization of DNNs as a tool for OTA computing applications. Simulation results demonstrate the efficiency of our approach while providing valuable insights.

C. NOTATION

From hereon, vectors are represented as bold lowercase letters. The Euclidean norm is denoted as $\|\cdot\|_2$ and the cardinality of a set or the absolute value of a real number, depending on the context, is denoted as $|\cdot|$. The expectation of a random variable is denoted as $\mathbb{E}[\cdot]$. Finally, the symbol $\odot$ is used to represent element-wise vector multiplication.

II. OTA COMPUTING PRELIMINARIES

We consider a wireless network consisting of K users, indexed as $k \in \mathcal{K} \triangleq \{1, 2, \ldots, K\}$, and a base station (BS) collocated with a server. In the following, the terms BS and server are used interchangeably. Each user possesses a measurement $x_k \in \mathbb{R}$, $\forall k \in \mathcal{K}$ to transmit to the BS. The task of the BS is to compute a desired function of interest, $\phi : \mathbb{R}^K \to \mathbb{R}$, which is a function of all the measurements x_k, $\forall k \in \mathcal{K}$. According to [2], it is possible to express any multivariate real-valued function as a composition of a finite sum of univariate functions. This means that there exists a collection of pre-processing functions, denoted as $f_k : \mathbb{R} \to \mathbb{R}$, $\forall k \in \mathcal{K}$, and a post-processing function $g : \mathbb{R} \to \mathbb{R}$, such that the original function can be expressed as

$$\phi(x_1, x_2, \ldots, x_K) = g\left(\sum_{k=1}^{K} f_k(x_k)\right). \tag{1}$$

Based on this idea and by exploiting the superposition property of a wireless MAC, devices can transmit their measurements x_k simultaneously by selecting an appropriate pre-processing function f_k. The transmitted signals are aggregated over-the-air, while subsequently the BS constructs the target function by applying the post-processing function g to the superimposed received signal. In a real wireless environment, with the presence of fading and additive noise, the estimation of the target function at the BS side is expressed as

$$\hat{\phi} = g\left(\sum_{k=1}^{K} h_k f_k(x_k) + n\right), \tag{2}$$

where $h_k \in \mathbb{C}$ denotes the channel coefficient of the k-th user. Hereinafter, it is clarified that the channel coefficients are treated as real numbers, since with perfect CSI at the transmitter, the magnitude of h_k can remain unchanged by only adjusting its phase, i.e., phase compensation [4]. It is also assumed that the devices' transmissions are well synchronized [7]. Moreover, n is additive white Gaussian noise (AWGN) at the receiver and follows the normal distribution $\mathcal{N}(0, \sigma_n^2)$[1] with zero mean and variance σ_n^2. By considering a peak average power constraint P for each device, the average transmit power must satisfy

$$\mathbb{E}\left[|f_k(x_k)|^2\right] \leq P, \quad \forall k \in \mathcal{K}, \tag{3}$$

where the expectation is taken with respect to the randomness of the measurement x_k. The statistical properties of x_k are arbitrary.

III. PROPOSED DNN OTA COMPUTING MECHANISM

The main objective of this work is the proper design of DNNs to approximate the pre-processing functions f_k, $\forall k \in \mathcal{K}$, and the post-processing function g. First, we assume a dedicated trained DNN to approximate each of the above functions. The trained parameters of the DNNs are denoted

[1]Although the noise n has both in-phase and quadrature components, the quadrature component is completely rejected when using an amplitude modulation (AM) coherent detection at the receiver [24]. As a consequence, n is real-valued and simply additive at the receiver.

as $\boldsymbol{w}_k$, $\forall k \in \mathcal{K}$, and $\boldsymbol{w}_g$, respectively. Accordingly, the parameterized approximations of the above functions are denoted as $f_{\boldsymbol{w}_k}$, $\forall k \in \mathcal{K}$, and $g_{\boldsymbol{w}_g}$. Taking these into account, the estimation $\hat{\phi}$ of the target function ϕ at the BS can be expressed as

$$\hat{\phi} = g_{\boldsymbol{w}_g}\left(\sum_{k=1}^{K} h_k f_{\boldsymbol{w}_k}(x_k; h_k) + n\right). \tag{4}$$

The rationale behind incorporating the channel coefficient h_k as an input to the function $f_{\boldsymbol{w}_k}$ will become clear later in this work. Also, note here the assumption of utilizing multiple DNNs, corresponding to each of the devices pre-processing functions and the server's post-processing function. The strategy that should be adopted for the effective design of such a system is not readily apparent. The challenge lies in the underlying interdependencies in the structure of the DNNs, considering that the devices and the server should somehow cooperate to approximate the target function with minimal error. Thus, the system should be analyzed from a holistic perspective, i.e., a unified loss function should be defined that aims to capture the system as a whole. To address this issue, we adopt a centralized training-decentralized execution approach, which will be explained in detail later in this section.

A. EXTENSION TO AN AUGMENTED VECTOR REPRESENTATION

Since we invoke the use of DNNs, there is no restriction in requiring that $f_{\boldsymbol{w}_k}$, $\forall k \in \mathcal{K}$, is scalar. As such, let $f_{\boldsymbol{w}_k} : \mathbb{R} \to \mathbb{R}^M$, $\forall k \in \mathcal{K}$, where M is the size of the output of $f_{\boldsymbol{w}_k}$, which we call the *augmented vector*. Also, let $g_{\boldsymbol{w}_g} : \mathbb{R}^M \to \mathbb{R}$. This strategy allows the representation of signals in a higher dimensional space, increasing the potential for more accurate reconstruction of the target function. However, to transmit the augmented vector via OTA computing, each entry of the vector should be aggregated over-the-air separately. To accomplish this, we consider an analog multi-carrier (MC) OTA computing implementation [7]. In the MC-OTA protocol, the available bandwidth is divided by the number of non-interfering subcarriers, which is considered to be equal to M for simplicity. Hence, each element of the augmented vector is aggregated over-the-air in a single orthogonal subcarrier. Consequently, the received signal at the BS side can be expressed as

$$\hat{\phi} = g_{\boldsymbol{w}_g}\left(\sum_{k=1}^{K} \boldsymbol{h}_k \odot f_{\boldsymbol{w}_k}(x_k; \boldsymbol{h}_k) + \boldsymbol{n}\right), \tag{5}$$

where $\boldsymbol{h}_k = (h_{k,1}, \ldots, h_{k,M}) \in \mathbb{R}^M$, with $h_{k,m}$ being the channel coefficient of user k in the m-th subcarrier. Also, $\boldsymbol{n} \sim \mathcal{N}(\boldsymbol{0}, \frac{\sigma_n^2}{M}\boldsymbol{I}_M)$, where $\boldsymbol{I}_M$ denotes the $M \times M$ identity matrix. Note that the variance of the noise is spread among the subcarriers.

For the sake of fairness between single-carrier OTA and MC OTA, it is essential to ensure not only that the total bandwidth is the same in both cases, but also that the energy for an OTA transmission is equivalent. Here, with the term OTA transmission we refer to the transmission of the measurement x_k, for an arbitrary user k. Since the bandwidth in the single-carrier OTA computing is W and the peak power constraint is P, the energy for an OTA transmission in the single-carrier case is $E = PT_s = P/W$, where $T_s = 1/W$ is the symbol duration. Accordingly, the energy for an MC OTA transmission case is given as

$$E_{\text{MC}} = \frac{P_{\text{MC}}}{\frac{W}{M}}, \tag{6}$$

where $P_{\text{MC}} = \sum_{i=1}^{M} P_i$ denotes the maximum power of MC OTA across all subcarriers. To enforce same energy levels between the MC and the single-carrier OTA computing transmissions, we have

$$\frac{P}{W} = \frac{P_{\text{MC}}}{\frac{W}{M}} \Rightarrow P_{\text{MC}} = \frac{P}{M}. \tag{7}$$

As a result, the average power constraint of a user for the MC OTA computing should satisfy

$$\mathbb{E}\left[\left\|f_{\boldsymbol{w}_k}(x_k; \boldsymbol{h}_k)\right\|_2^2\right] \leq \frac{P}{M}, \quad \forall k \in \mathcal{K}. \tag{8}$$

It is clarified that the application of augmented vector representation, as described previously, can also be extended to MIMO OTA systems, such as in [6].

B. CENTRALIZED TRAINING

First, a single cDNN is considered and its training parameters are represented by $\boldsymbol{w} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_K, \boldsymbol{w}_g)$. The cDNN resides in the server whose task is to train it. The structure of the cDNN is depicted in Fig. 1. The concatenated feature vector which is fed as input to the cDNN is described as $\boldsymbol{v} = (x_1, \boldsymbol{h}_1, x_2, \boldsymbol{h}_2, \ldots, x_K, \boldsymbol{h}_K) \in \mathbb{R}^{K+KM}$. Following that, each tuple $\{x_k, \boldsymbol{h}_k\}_{k\in\mathcal{K}}$ is passed through a separate set of hidden layers (HLs), represented by the training parameters $\boldsymbol{w}_k$, $\forall k \in \mathcal{K}$. Next, the output of each set of HLs is passed through the customized activation function $A : \mathbb{R}^M \to \mathbb{S}$, where $\mathbb{S}$ is the surface of the M-dimensional sphere with radius $\sqrt{\frac{P}{M}}$, with

$$A(\boldsymbol{z}) = \sqrt{\frac{P}{M}}\frac{\boldsymbol{z}}{\|\boldsymbol{z}\|_2}. \tag{9}$$

This function is chosen to satisfy the average power constraint of the devices in (8), while it is easy to see that $\|A(\boldsymbol{z})\|_2^2 = \frac{P}{M}$. Therefore, the considered activation function is the underlying mechanism that regulates the power of the signal to facilitate OTA computing and also performs the power allocation among the subcarriers. While theoretically selecting the function $A'(\boldsymbol{z}) \triangleq A(\boldsymbol{z}) \odot \tanh(\boldsymbol{z})$ may seem more prudent, as it does not necessarily enforce equality in (8), experimental observations indicate that it achieves slightly lower performance compared to using $A(\boldsymbol{z})$. This is probably due to its complicated form as an activation

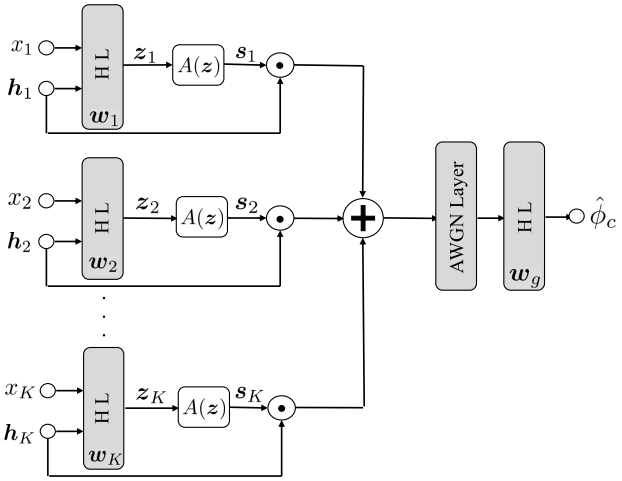

FIGURE 1. cDNN Architecture.

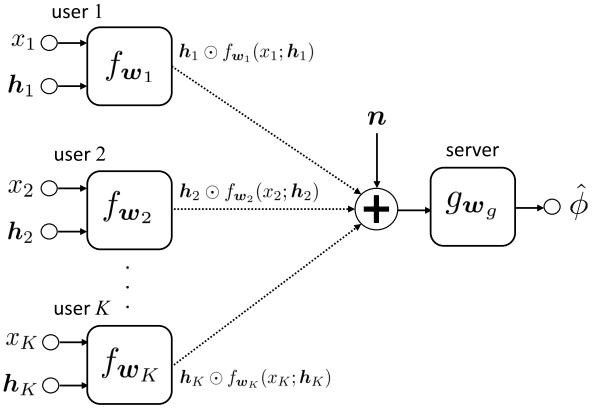

FIGURE 2. Decentralized DNN OTA computing execution.

function for a neural network. However, in case where $M = 1$, selecting the activation function $A'(z)$, which is reduced to $\sqrt{P}\tanh(z)$, works fine. On the contrary, the function $A(z)$ fails completely when $M = 1$, since it only outputs the values $\pm\sqrt{P}$.

Up to this point, the purpose of this part of the cDNN is to provide an interpretation of the functions $f_{\boldsymbol{w}_k}$, $\forall k \in \mathcal{K}$. In what follows, each augmented vector $\boldsymbol{s}_k$ is multiplied element-wise with the respective input $\boldsymbol{h}_k$, $\forall k \in \mathcal{K}$. Next, a noise layer is added in the resulting sum of products, $\sum_{k\in\mathcal{K}} \boldsymbol{h}_k \odot \boldsymbol{s}_k$. This part of the cDNN aims to mimic the over-the-air transmission in a wireless MAC. To this end, a set of HLs is included, represented by the training parameters $\boldsymbol{w}_g$, aiming to interpret the function $g_{\boldsymbol{w}_g}$. The ultimate goal of the cDNN is to provide an approximation of the target function ϕ through its final output, denoted as $\hat{\phi}_c = \hat{\phi}_c(\boldsymbol{v}, \boldsymbol{w})$.

To train the cDNN, a training dataset is first generated, denoted as $\{\boldsymbol{v}_t, \phi(\boldsymbol{x}_t)\}_{t\in\mathcal{T}}$, where $\mathcal{T}$ is the set of all training samples and $\boldsymbol{x} = (x_1, x_2, \ldots, x_K)$. Hence, $\boldsymbol{v}_t$ represents the t-th input sample and $\phi(\boldsymbol{x}_t)$ is the value of the target function given the t-th input sample. Following that, the loss function of the cDNN is selected as the MSE between the target function and the output of the cDNN, i.e.,

$$L\left(\phi, \hat{\phi}_c\right) = \frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}}\left(\phi(\boldsymbol{x}_t) - \hat{\phi}_c(\boldsymbol{v}_t, \boldsymbol{w})\right)^2. \tag{10}$$

Finally, the server trains the cDNN by using standard back-propagation to minimize the loss function, i.e., find $\boldsymbol{w}^* = \underset{\boldsymbol{w}}{\operatorname{argmin}}\, L(\phi, \hat{\phi}_c)$.

C. DECENTRALIZED DNN OVER-THE-AIR EXECUTION (DOTACE)

The previous process described the centralized training of a cDNN, which aims to capture the characteristics of OTA computing and approximate the pre- and post-processing functions. However, to practically use DNNs for this task in the context of a wireless network, each device, as well as the BS, should possess a dedicated trained DNN which individually approximates the pre- and post-processing functions, respectively. In this direction, the procedure of the DOTACE is described below.

First, after the training of the cDNN, the server sends the trained parameters $\boldsymbol{w}_k$ to each device k. Taking this into consideration, each user owns a trained local DNN, which is denoted as $f_{\boldsymbol{w}_k}$. The architecture of the local DNN for user k is equivalent to the corresponding part of the cDNN, including the activation function $A(\cdot)$, implying that $f_{\boldsymbol{w}_k}(x_k; \boldsymbol{h}_k) = \boldsymbol{s}_k$. In other words, when user k forward-passes its local DNN given the input vector $(x_k, \boldsymbol{h}_k)$, the corresponding output is equal to $\boldsymbol{s}_k$, which is the augmented vector. Similarly, the server uses the parameters $\boldsymbol{w}_g$ to create another DNN representing the function $g_{\boldsymbol{w}_g}$. At this point, since all users and the server own an approximator for the functions f_k, $\forall k \in \mathcal{K}$ and g, the OTA computing technique can be implemented, while this process is depicted in Fig. 2. It should be highlighted that unlike the cDNNs structure, where $\boldsymbol{s}_k$ is multiplied with $\boldsymbol{h}_k$, $\forall k \in \mathcal{K}$, in the DOTACE there is no need to adopt such an approach. The multiplication with the channel coefficients and the summation of all signals will take place over-the-air via the wireless MAC. The output of the DOTACE is denoted as $\hat{\phi}$, which is an estimation of the target function ϕ.

It is now evident that the cDNN has been designed to allow its decentralized execution. Its tailored architecture and the incorporation of OTA computing principles throughout the design aim to guarantee the effectiveness of the DOTACE. To solidify this claim, we present the following remark.

Remark 1: In the absence of AWGN, meaning that $n = 0$ and the AWGN layer is discarded from the cDNN, it holds $\hat{\phi} = \hat{\phi}_c$, i.e., the cDNN and DOTACE have the same estimation output.

This remark indicates that in the absence of AWGN, the outputs of the cDNN and the DOTACE are identical. This is easy to verify. Given an input vector $\boldsymbol{v}$, the forward-pass of the cDNN yields identical result with the DOTACE. Alternatively, if the two implementations are viewed as computational graphs, these graphs are identical. Moreover, even in the presence of AWGN, it is straightforward to see that the outputs are the same in expectation, i.e., $\mathbb{E}_n[\hat{\phi}] = \mathbb{E}_n[\hat{\phi}_c]$. This observation is very encouraging, since

it suggests that if the cDNN provides an accurate estimation of the target function ϕ, so does the DOTACE.

Remark 2: The DOTACE requires only transmitter CSI.

Recall that the feature input vector of each local DNN $f_{\boldsymbol{w}_k}$ is $(x_k, \boldsymbol{h}_k), \forall k \in \mathcal{K}$. This implies that user k only requires information about its own channel $\boldsymbol{h}_k$, while neither the remaining users, $\mathcal{K} \setminus \{k\}$, nor the BS need this information. This fact is of paramount importance, as it reduces the pilot signaling overhead related with the CSI acquisition. Moreover, the requirement for the BS to collect all users' CSI and calculate the pre-processing and post-processing functions, which subsequently are sent back to the users [4], is eliminated.

IV. ORACLE BENCHMARK

A. DESCRIPTION

In order to assess and evaluate the performance of DOTACE, it is crucial to provide a benchmark. This benchmark will serve as a reference point for comparing and measuring the effectiveness of DOTACE. In this context, we propose the utilization of an OB, which is outlined as follows. Consider the existence of a "superuser" with knowledge of all users' signals $x_1, x_2, \ldots, x_K$, while being able to compute the target function $\phi(\boldsymbol{x})$ instantly and losslessly. Afterwards, the superuser transmits ϕ to the server via a wireless analog transmission, thus the received signal at the BS is expressed as

$$y = \frac{1}{a}\left(\sqrt{P_s}h\phi + n\right), \tag{11}$$

where h is the complex channel coefficient experienced by the superuser, P_s is the transmit power, and $a > 0$ is the BS post-processing scaling factor. For a fair comparison, we assume that the peak power constraint of the superuser is KP, i.e., equal to the sum of the users' peak power constraints. Hence, the average transmit power should satisfy

$$\mathbb{E}\left[P_s\phi^2\right] \leq KP. \tag{12}$$

The expected MSE between y and ϕ, defined as

$$F \triangleq \mathbb{E}\left[(\phi - y)^2\right], \tag{13}$$

is used as the OB. The term oracle originates from the concept of an imaginary scenario, where a superuser possesses complete knowledge of all other users' messages. Obviously, the performance of the OB in terms of MSE cannot be surpassed by any OTA computing-based scheme, since the superuser has knowledge of the target function prior to its transmission to the BS. Therefore, the OB serves as an MSE lower bound not only for the proposed DOTACE, but in general for any OTA computing-based implementation.

B. MINIMUM EXPECTED MSE

In this subsection, the minimum value of the expected MSE between the target value and the BS's estimation is derived. First, we normalize ϕ as

$$\tilde{\phi} = \frac{\phi - \mu_\phi}{\sigma_\phi}, \tag{14}$$

where μ_ϕ is the mean and σ_ϕ the standard deviation of ϕ. As such, $\tilde{\phi}$ is a random variable with zero mean and unit variance. Next, by setting the transmit power as $P_s = KP$, it is easy to verify that $\mathbb{E}[P_s\tilde{\phi}^2] = KP$, which satisfies the constraint in (12). With this in mind, and taking into account (11), the received signal at the BS is given by

$$y_r = \frac{1}{a}\left(\sqrt{KP}h\tilde{\phi} + n\right). \tag{15}$$

To recover the signal ϕ assuming its statistics are known, the BS denormalizes y_r and calculates the estimation

$$\begin{aligned}\hat{y} &= \sigma_\phi y_r + \mu_\phi \\ &= \frac{\sigma_\phi}{a}\left(\sqrt{KP}h\frac{\phi - \mu_\phi}{\sigma_\phi} + n\right) + \mu_\phi.\end{aligned} \tag{16}$$

The objective of the BS is to find the optimal parameter a, such that the expected MSE between ϕ and $\hat{y}$ is minimized, i.e.,

$$a^* = \underset{a>0}{\operatorname{argmin}}\ F(a), \tag{17}$$

where

$$\begin{aligned}F(a) &= \mathbb{E}_{\phi,n}\left[\left(\phi - \hat{y}\right)^2\right] \\ &= \mathbb{E}_{\phi,n}\left[\left(\phi - \frac{\sigma_\phi}{a}\left(h\sqrt{KP}\frac{\phi - \mu_\phi}{\sigma_\phi} + n\right) - \mu_\phi\right)^2\right].\end{aligned} \tag{18}$$

Note that we assume perfect CSI and, thus, h is known, while the expectation in (18) is taken with respect to the randomness of the signal ϕ and the noise n.

Proposition 1: The optimal a of problem (17) is given as

$$a^* = \frac{h^2KP + \sigma_n^2}{h\sqrt{KP}} \tag{19}$$

and the minimum achieved MSE of the OB is equal to

$$F(a^*) = \frac{\sigma_\phi^2}{\gamma + 1}, \tag{20}$$

where $\gamma \triangleq \frac{h^2KP}{\sigma_n^2}$ reflects the signal-to-noise ratio (SNR).

Proof: To deal with (17), we first simplify the objective function. After some algebraic manipulations, the objective function is given by

$$F(a) = \sigma_\phi^2\left(1 - \frac{h\sqrt{KP}}{a}\right)^2 + \frac{\sigma_\phi^2\sigma_n^2}{a^2}. \tag{21}$$

It is easy to observe that the first derivative of F with respect to a is equal to

$$\frac{dF(a)}{da} = 2\sigma_\phi^2\left(\frac{h\sqrt{KP}}{a^2} - \frac{h^2KP + \sigma_n^2}{a^3}\right). \tag{22}$$

The extrema of (21) are known to be achieved at the roots of (22). It is straightforward to see that the root of (22), which is equal to

$$a_{\text{ext}} = \frac{h^2KP + \sigma_n^2}{h\sqrt{KP}}, \tag{23}$$

is unique, which implies that the extremum is global, since $F(a)$ is continuous and its domain is a continuous interval. If we rewrite (22) more conveniently as

$$\frac{dF(a)}{da} = -\frac{2\sigma_\phi^2}{a^2}\left(\frac{h^2KP+\sigma_n^2}{a} - h\sqrt{KP}\right), \tag{24}$$

it is obvious that for $a < a_{\text{ext}}$ and for $a > a_{\text{ext}}$ the sign of (24) is negative and positive, respectively, meaning that the extremum achieved at $a = a_{\text{ext}}$ is the global minimum due to the monotonicity of F, thus $a^* = a_{\text{ext}}$. After substituting (23) in (21) and performing some algebraic manipulations, the minimum of F can be calculated as in (20), which completes the proof. ■

Driven by Proposition 1, we present the following remark.

Remark 3: In very high and very low SNR regimes, i.e., $\gamma \to \infty$ and $\gamma \to 0$, the minimum expected value of the MSE given by (20) becomes

$$\lim_{\gamma\to\infty} F(a^*) = 0 \tag{25}$$

and

$$\lim_{\gamma\to 0} F(a^*) = \sigma_\phi^2. \tag{26}$$

As expected, for very high SNR, the average MSE is zero. Accordingly, for very low SNR, the result in (26) coincides with that of [25]. This indicates that the optimal estimation in this case, which results in minimum MSE, is determined by the variance of the target signal.

V. SIMULATIONS

To evaluate the performance of the proposed DOTACE framework, we first consider a wireless network with $K = 2$ users, while their measurements x_1, x_2 are independent and identically distributed and follow uniform distribution in $[-\sqrt{3}, \sqrt{3}]$. Therefore, x_1 and x_2 have zero mean and unit variance. As in [18], we choose the following target function

$$\phi_{\text{A}}(x_1, x_2) = \frac{1}{\phi_{\text{A,max}}}(x_1 + x_1x_2 + x_2)^2, \tag{27}$$

where $\phi_{\text{A,max}} = (2\sqrt{3}+3)^2$. As such, $\phi_{\text{A}} \in [0, 1]$. After some computations, we have calculated the mean and standard deviation of ϕ_{A} as $\mu_{\phi_{\text{A}}} = 0.0718$ and $\sigma^2_{\phi_{\text{A}}} = 0.0145$. In general, if calculating the mean and variance of a function ϕ becomes intractable, one can resort to using the sampled mean and variance as an alternative. Moreover, the channel coefficients between the users and the BS are independent and identically distributed and follow the Rayleigh distribution with scale parameter $1/\sqrt{2}$. We also set the peak transmit power of the users as $P = 1$ W. As a result, the average received SNR of a single user transmission in a single carrier is equal to $-20\log_{10}(\sigma_n)$ dB.

At first glance, one might argue that the nomographic form of ϕ_{A} in (27) can be described by the pre-processing functions $f_k(x_k) = \ln(x_k + 1)$, $k \in \{1, 2\}$, and the post-processing function $g(r) = (\exp(r) - 1)^2$, where $r = \sum_{k=1}^{2} f_k(x_k)$ is the ideally received signal at the BS. Unfortunately, this representation does not hold for $x_k \leq -1$. Therefore, the assumed pre-processing functions cannot be used to describe ϕ_{A}, which triggers the exploration of more sophisticated functions to achieve this goal. Beside this, with the inclusion of fading, noise, and power constraints, representing ϕ_{A} in a (quasi) nomographic form would become even more challenging, if not impossible. In contrast, the proposed DNN mechanism is not affected by the domain of the measurements and also satisfies the considered power constraint by design.

To evaluate the performance of DOTACE for a larger number of users, we consider the target function

$$\phi_{\text{B}}(x_1, x_2, \ldots, x_K) = \sum_{k=1}^{K} x_k + \prod_{k=1}^{K} x_k. \tag{28}$$

Again, it is not obvious that ϕ_{B} can be represented in a nomographic closed-form. Since no current method can be used as a benchmark for these functions, to assess the performance of the proposed DOTACE scheme, we conduct a comparative analysis against the OB scheme.

A. CDNN SETUP

Next, the structure of the cDNN is outlined, noting that the cDNN hyperparameters were obtained through simulation experiments and fine-tuning. For the HLs representing the functions $f_{\boldsymbol{w}_k}$, $\forall k \in \mathcal{K}$, we used two fully connected feed-forward layers with 256 neurons and ReLU activation, followed by a layer of M nodes passing through the activation function $A(\cdot)$. Also, for the HLs representing $g_{\boldsymbol{w}_g}$, we used an input layer consisting of M nodes and a single fully connected feed forward layer with 256 neurons followed by a single node layer with no activation, which is the final output. For the training dataset, $|\mathcal{T}| = 5 \times 10^4$ samples were generated. It was observed that further increasing the number of the cDNN's training parameters or the size of the training dataset does not result in substantial improvement. Moreover, the optimization algorithm used to train the cDNN is the Adam [26], with the first and second moments of the decay rate set to 0.99 and 0.999, respectively. In addition, the learning rate was set to 0.001 and the batch size to 64. The training process included 200 epochs, although it was observed through experimentation that a steady state can be achieved with fewer epochs. Finally, in the testing phase, 3×10^4 samples were used.[2]

B. RESULTS AND DISCUSSION

In Fig. 3, the performance of the cDNN, DOTACE, and the OB are demonstrated for the function ϕ_{A}. The evaluation metric is the average MSE, which was calculated for different values of the average received SNR. It should be noted that a dedicated cDNN was trained for each SNR value. In practical implementations, alterations in environmental factors, e.g.,

[2]In the spirit of reproducible research, the code used for the simulations is available at: https://github.com/paulBooz/OTA-DNN.

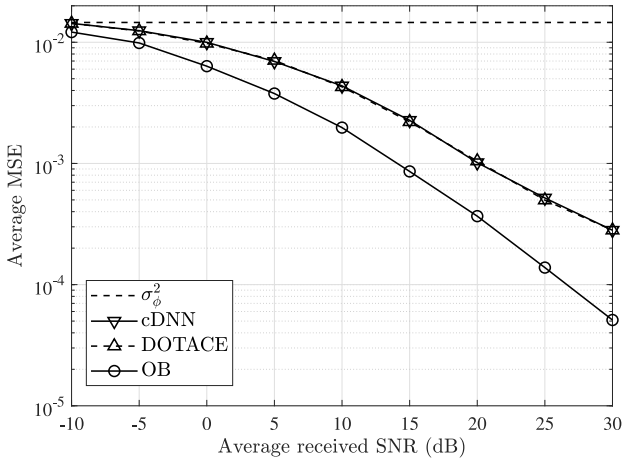

FIGURE 3. Average MSE versus average received SNR for the target function ϕ_A.

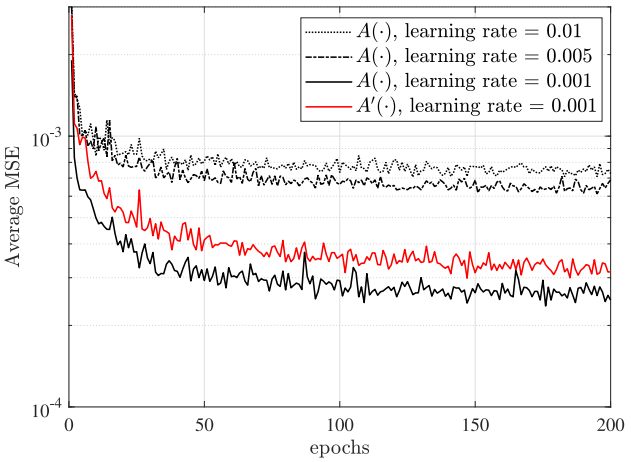

FIGURE 4. Average MSE versus training epochs for the target function ϕ_A.

channel distribution, number of users, would necessitate the retraining of the cDNN. The task of tackling the dynamic changes in the context of DNNs is left for future work, as it remains a distinct and standalone open challenge [27]. Moreover, for this figure we consider flat-fading channels in an analog MC system with $M = 4$ subcarriers. The flat-fading assumption ensures that the comparison with OB, where a single carrier is considered, is fair. Also, the OB is constructed by substituting the optimal scaling parameter a^* from (19) into (15). Finally, recall that the worst case scenario of the average MSE is equal to σ_ϕ^2, as outlined in Remark 3.

First, it is observed that the cDNN and DOTACE illustrate identical performance, validating Remark 1. This fact again corroborates the effectiveness of the centralized training-decentralized execution approach. Moreover, in a very low SNR regime, all approaches fail to provide a good estimation and converge to the worst case MSE. As the SNR increases, both the DOTACE and the OB significantly outperform the MSE worst case scenario. Also, the gap between DOTACE and OB increases as the SNR reaches higher levels. This is reasonable, since the average MSE of OB asymptotically tends to zero, as suggested in Remark 3. Furthermore, the MSE of DOTACE is expected to reach a floor, possibly due to the inherent approximation error associated with the DNNs. However, recall that the OB reflects a hypothetical scenario that is unattainable in practice.

In Fig 4, we illustrate the evolution of the training loss over multiple epochs, with a fixed received SNR of 30 dB. The purpose of this figure is to provide insights into the selection of the learning rate and the activation function of the HLs. This simulation justifies the choice of a learning rate of 0.001, as it consistently yields the lowest loss compared to other learning rate values. Additionally, we observe that using the activation function $A(\cdot)$ from equation (9) leads to a reduction in loss compared to the alternative choice, $A'(\cdot)$, which justifies the adoption of the former.

In Fig. 5, different sizes M of the augmented vector are demonstrated. It is obvious that the MSE is reduced as M

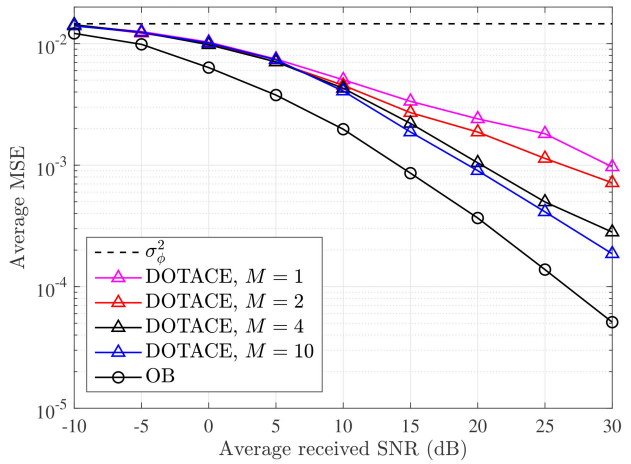

FIGURE 5. Average MSE versus average received SNR for various sizes of the augmented vector for the target function ϕ_A.

increases, with a notable difference when sweeping M from 2 to 4. The case $M = 1$ corresponds to a single-carrier transmission. This experiment verifies the efficiency of using higher dimensional vectors to represent the output of the pre-processing functions. However, it should be clarified that the transmission time grows proportionally with M, since M also defines the number of subcarriers in the MC DOTACE. This is related to the analog symbol duration, which in the case of MC transmission is equal to $\frac{M}{W}$. It is evident that the value $M = 4$ seems suitable for achieving a satisfactory MSE-transmission delay trade-off.

In the continue, we evaluate the performance of DOTACE for a larger number of users on the function ϕ_B. First, it is easy to verify that $\sigma_{\phi_B}^2 = K + 1$. For this reason, we choose to evaluate the metric of the normalized expected MSE, i.e., $\frac{\mathbb{E}[\text{MSE}]}{K+1}$, implying that the worst case performance occurs when the considered metric is equal to 1, for any choice of K. For this simulation the cDNN was trained for 200 epochs, an augmented vector size of $M = 4$ was chosen, and the SNR was set to 30 dB. In Fig. 6, the normalized average MSE versus the number of users is demonstrated.

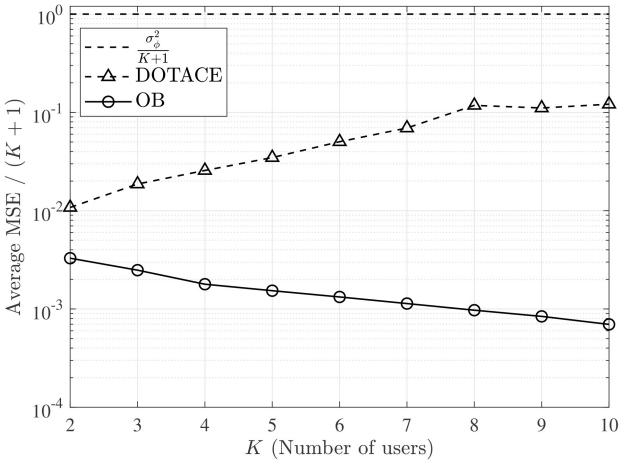

FIGURE 6. Normalized average MSE versus number of users for the target function ϕ_{B}.

It can be observed that as K increases, the normalized MSE of DOTACE also increases. This behavior is probably related to insufficient knowledge of user CSI combined with the challenge of representing ϕ_{B} in nomographic form. Recall that each user has knowledge of its individual CSI. Thus, as the number of users increases, it becomes increasingly difficult to find suitable pre- and post-processing functions, partly due to the detrimental effects of limited CSI knowledge. This aspect presents potential avenues for future exploration concerning the scalability of DOTACE. Nevertheless, even when considering $K = 10$ users, the normalized MSE of DOTACE is an order of magnitude lower than the worst-case scenario. This observation suggests that DOTACE does not rely on random guesses, but rather learns to imperfectly represent ϕ_{B} in nomographic form.

VI. CONCLUSION

In this work, the use of a deep learning mechanism is invoked to approximate the pre- and post-processing functions of OTA computing. The proposed mechanism is based on centralized training, which encapsulates the characteristics of a wireless MAC, and DOTACE, which allows the users and the server to independently approximate the pre- and post-processing functions, respectively. The vision is to facilitate the approximation of any function through OTA computing, without limiting its use to specific functions that are currently handled. Significantly, DOTACE requires only transmitter CSI. Our study also focused on the introduction of a benchmark scheme that serves as a reference point on the computational distortion of any OTA computing target function. The numerical results revealed both the potential and the limitations of the proposed mechanism. To further unveil the merits of DOTACE, the inclusion of complete CSI knowledge both at the transmitter and the receiver, in the expense of additional overhead, can be an avenue for future research. To pave the way for practical applications, exploration of techniques aimed at enhancing the scalability of DOTACE can also be pursued in this direction.

REFERENCES

[1] B. Nazer and M. Gastpar, "Computation over multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3498–3516, Oct. 2007.

[2] M. Goldenbaum and S. Stanczak, "Robust analog function computation via wireless multiple-access channels," *IEEE Trans. Commun.*, vol. 61, no. 9, pp. 3863–3877, Sep. 2013.

[3] M. Goldenbaum, H. Boche, and S. Stańczak, "Analyzing the space of functions analog-computable via wireless multiple-access channels," in *Proc. 8th Int. Symp. Wireless Commun. Syst.*, 2011, pp. 779–783.

[4] W. Liu, X. Zang, Y. Li, and B. Vucetic, "Over-the-air computation systems: Optimization, analysis and scaling laws," *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5488–5502, Aug. 2020.

[5] L. Chen, N. Zhao, Y. Chen, F. R. Yu, and G. Wei, "Over-the-air computation for IoT networks: Computing multiple functions with antenna arrays," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 5296–5306, Dec. 2018.

[6] D. Wen, G. Zhu, and K. Huang, "Reduced-dimension design of MIMO over-the-air computing for data aggregation in clustered IoT networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5255–5268, Nov. 2019.

[7] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 491–506, Jan. 2020.

[8] M. M. Amiri and D. Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," *IEEE Trans. Signal Process.*, vol. 68, pp. 2155–2169, 2020.

[9] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Trans Wireless Commun.*, vol. 19, no. 3, pp. 2022–2035, Mar. 2020.

[10] C. Xu, S. Liu, Z. Yang, Y. Huang, and K.-K. Wong, "Learning rate optimization for federated learning exploiting over-the-air computation," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3742–3756, Dec. 2021.

[11] S. Wang, Y. Hong, R. Wang, Q. Hao, Y.-C. Wu, and D. W. K. Ng, "Edge federated learning via unit-modulus over-the-air computation," *IEEE Trans. Commun.*, vol. 70, no. 5, pp. 3141–3156, May 2022.

[12] T. Sery, N. Shlezinger, K. Cohen, and Y. C. Eldar, "Over-the-air federated learning from heterogeneous data," *IEEE Trans. Signal Process.*, vol. 69, pp. 3796–3811, 2021.

[13] G. Shi, S. Guo, J. Ye, N. Saeed, and S. Dang, "Multiple parallel federated learning via over-the-air computation," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 1252–1264, 2022.

[14] X. Fan, Y. Wang, Y. Huo, and Z. Tian, "Joint optimization of communications and federated learning over the air," *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, pp. 4434–4449, Jun. 2022.

[15] N. A. Mitsiou, P. S. Bouzinis, P. D. Diamantoulakis, R. Schober, and G. K. Karagiannidis, "Accelerating distributed optimization via over-the-air computing," *IEEE Trans. Commun.*, vol. 71, no. 9, pp. 5565–5579, Sep. 2023.

[16] Z. Lin, Y. Gong, and K. Huang, "Distributed over-the-air computing for fast distributed optimization: Beamforming design and convergence analysis," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 1, pp. 274–287, Jan. 2023.

[17] D. Wen, X. Jiao, P. Liu, G. Zhu, Y. Shi, and K. Huang, "Task-oriented over-the-air computation for multi-device edge AI," *IEEE Trans Wireless Commun.*, vol. 23, no. 3, pp. 2039–2053, Mar. 2024.

[18] S. Limmer, J. Mohammadi, and S. Stańczak, "A simple algorithm for approximation by nomographic functions," in *Proc. 53rd Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, 2015, pp. 453–458.

[19] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep over-the-air computation," in *Proc. IEEE Global Commun. Conf.*, 2020, pp. 1–6.

[20] T. O'Shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Trans. Cogn. Commun. Netw.*, vol. 3, no. 4, pp. 563–575, Dec. 2017.

[21] D. Wu, M. Nekovee, and Y. Wang, "Deep learning-based autoencoder for m-user wireless interference channel physical layer design," *IEEE Access*, vol. 8, pp. 174679–174691, 2020.

[22] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in OFDM systems," *IEEE Wireless Commun. Lett.*, vol. 7, no. 1, pp. 114–117, Feb. 2018.

[23] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for interference management," *IEEE Trans. Signal Process.*, vol. 66, no. 20, pp. 5438–5453, Oct. 2018.
[24] S. Haykin, *Communication Systems*. Hoboken, NJ, USA: Wiley 2008.
[25] T. M. Cover, *Elements of Information Theory*. Hoboken, NJ, USA: Wiley, 1999.
[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[27] M. Akrout, A. Mezghani, E. Hossain, F. Bellili, and R. W. Heath, "From multilayer perceptron to GPT: A reflection on deep learning research for wireless physical layer," 2023, *arXiv:2307.07359*.

PAVLOS S. BOUZINIS received the Diploma (five years) and Ph.D. degrees in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 2019 and 2023, respectively, where he was a member of the Wireless Communications and Information Processing Group. He is currently works as a Researcher with MetaMind Innovations P.C. His main research interests include machine learning, optimization, wireless communication networks, and intrusion detection systems.

NIKOS G. EVGENIDIS (Graduate Student Member, IEEE) received the Diploma (five years) degree in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 2022, where he is currently pursuing the Ph.D. degree. He is also a member of the Wireless and Communications and Information Processing Group. His major research interests include semantic communications, over-the-air computing, non-orthogonal multiple access, machine learning, and optimization theory.

NIKOS A. MITSIOU (Graduate Student Member, IEEE) received the Diploma (five years) degree in electrical and computer engineering from the Aristotle University of Thessaloniki, Greece, in 2021, where he is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering. He is a member of the Wireless and Communications and Information Processing Group. His research interests include optimization theory, and machine learning with application to wireless networks. He was an Exemplary Reviewer of the IEEE WIRELESS COMMUNICATIONS LETTERS in 2022 (top 3% of reviewers).

SOTIRIS A. TEGOS (Senior Member, IEEE) received the Diploma (five years) and Ph.D. degrees from the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2017 and 2022, respectively. Since 2022, he has been a Postdoctoral Fellow with the Wireless Communications and Information Processing Group, Aristotle University of Thessaloniki, and the Department of Applied Informatics, University of Macedonia, Thessaloniki. Since 2023, he has been also a Postdoctoral Fellow with the Department of Electrical and Computer Engineering, University of Western Macedonia, Kozani, Greece. In 2018, he was a Visiting Researcher with the Department of Electrical and Computer Engineering, Khalifa University, Abu Dhabi, UAE. His current research interests include multiple access in wireless communications, optical wireless communications, and reconfigurable intelligent surfaces. He is a Working Group Member of the Newfocus COST Action "European Network on Future Generation Optical Wireless Communication Technologies." He serves as an Editor for IEEE COMMUNICATIONS LETTERS. He received the Best Paper Award in 2023 Photonics Global Conference. He was an Exemplary Reviewer of IEEE WIRELESS COMMUNICATIONS LETTERS in 2019 and 2022 (top 3% of reviewers).

PANAGIOTIS D. DIAMANTOULAKIS (Senior Member, IEEE) received the Diploma (five years) and Ph.D. degrees from the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2012 and 2017, respectively, where he has been a Postdoctoral Fellow with Wireless Communications and Information Processing Group since 2017. Since 2021, he has been a Visiting Assistant Professor with the Key Laboratory of Information Coding and Transmission, Southwest Jiaotong University, Chengdu, China. Since 2022, he has been also the Principal Investigator of a national research project, which is being implemented at the Department of Applied Informatics, University of Macedonia. His research interests include optimization theory and applications in wireless networks, optical wireless communications, and goal-oriented communications. He is a Working Group Member of the Newfocus COST Action "European Network on Future Generation Optical Wireless Communication Technologies." He serves as an Editor of IEEE OPEN JOURNAL OF THE COMMUNICATIONS SOCIETY, *Physical Communications* (Elsevier), and *Communications and Networks* (Frontiers), from 2018 to 2023, he has been an Editor of IEEE WIRELESS COMMUNICATIONS LETTERS. He was also an Exemplary Editor of the IEEE WIRELESS COMMUNICATIONS LETTERS in 2020, and an Exemplary Reviewer of the IEEE COMMUNICATIONS LETTERS in 2014 and the IEEE TRANSACTIONS ON COMMUNICATIONS in 2017 and 2019 (top 3% of reviewers).

GEORGE K. KARAGIANNIDIS (Fellow, IEEE) is currently a Professor with the Electrical and Computer Engineering Department, Aristotle University of Thessaloniki, Greece, and the Head of Wireless Communications and Information Processing Group. He is also a Faculty Fellow with the Cyber Security Systems and Applied AI Research Center, Lebanese American University. His research interests are in the areas of wireless communications systems and networks, signal processing, optical wireless communications, wireless power transfer and applications and communications and signal processing for biomedical engineering. Recently, he received the three prestigious awards: The 2021 IEEE ComSoc RCC Technical Recognition Award, the 2018 IEEE ComSoc SPCE Technical Recognition Award, and the 2022 Humboldt Research Award from Alexander von Humboldt Foundation. He is one of the highly-cited authors across all areas of Electrical Engineering, recognized from Clarivate Analytics as Highly-Cited Researcher from 2015 to 2023. He was an Editor in several IEEE journals and from 2012 to 2015 he was an Editor-in Chief of IEEE COMMUNICATIONS LETTERS. From September 2018 to June 2022 he served as an Associate Editor-in Chief for IEEE OPEN JOURNAL OF COMMUNICATIONS SOCIETY. He is currently an Editor-in-Chief of IEEE TRANSACTIONS ON COMMUNICATIONS.