ChannelComp: A general framework for computing by digital communication

SEYED SAEED RAZAVIKIA

Licentiate Thesis in Electrical Engineering
KTH Royal Institute of Technology
Stockholm, Sweden 2023

KTH Royal Institute of Technology
School of Electrical Engineering and Computer Science
Division of Network and Systems Engineering
SE-10044 Stockholm
Sweden

TRITA-EECS-AVL-2023:75
ISBN: 978-91-8040-741-0

Akademisk avhandling som med tillstånd av Kungl Tekniska högskolan framlägges till offentlig granskning för avläggande av Teknologie licentiatexamen i elektroteknik 2023-11-20 i E32, Osquars backe 2, E-huset, huvudbyggnaden, Lindstedtsvägen 3, floor 3.

Tryck: Universitetsservice US AB

Abstract

The imminent Internet of Things, fueled by 6G networks and machine learning technologies, is set to shift wireless communication to machine-centric paradigms, revolutionizing sectors such as healthcare or industrial automation through efficient data handling. However, this connectivity boom poses challenges, including straining existing communication systems due to increased data traffic and computational demands.

Over-the-air computation (OAC) presents a feasible solution, allowing the summation of transmitted signals at a common receiver through analog amplitude modulation. Designed to enable concurrent data collection and computation at the network edge, OAC seeks to lessen the central system burden, reducing latency and energy usage while enabling real-time analytics. This approach is particularly beneficial for federated learning, a machine learning technique that operates across decentralized devices. However, OAC's dependence on analog communication poses notable challenges, including signal distortion during transmission and the limited availability of devices supporting analog modulations. Digital modulation is a preferable alternative, recognized for its excellent channel correction capabilities and broad acceptance in modern wireless devices. Nevertheless, its integration into OAC is perceived as a significant hurdle, with overlapping digitally modulated signals threatening the fundamental concept of simultaneous data collection and computation.

The first part of the thesis provides an overview of communication systems, specifically focusing on the relevant OAC methodologies for analog and digital parts and their application in ML, particularly in training federated learning models. Subsequently, an exhaustive literature review concerning analog OAC techniques is undertaken, identifying existing limitations within this domain. The central thrust of our research is then introduced, proposing an innovative digital OAC approach along with a fresh perspective on the communication systems models designed for executing the computation. The chapter concludes with a summary of the principal contributions of each paper included within the thesis.

In the second part, we introduce ChannelComp, a groundbreaking computing approach compatible with current digital communication systems, including smartphones and IoT devices. A detailed analysis of ChannelComp's functions reveals how it enables digital modulation schemes to perform computations, addressing a critical gap in previous research. Moreover, introducing pre-coders designed for function computation over the multiple access channel, combined with a feasibility optimization problem framework, allows for seamless integration with current systems. Compared to OAC, restricted to analog modulations, ChannelComp exhibits broader computational capabilities and adherence to strict computation time constraints, thus showcasing its robust potential for future massive machine-type communications. This innovative method signifies a promising direction toward sustainable and efficient future wireless communication.

Sammanfattning

Den nära förestående Internet of Things, drivet av 6G-nätverk och maskininlärningsteknologier, är på väg att förändra trådlös kommunikation till maskincentrerade paradigm, revolutionerande sektorer som hälso- och sjukvård samt industriell automatisering genom effektiv datahantering. Dock medför denna uppkopplingsboom utmaningar, inklusive påfrestningar på befintliga kommunikationssystem på grund av ökad datatrafik och beräkningsbehov.

Over-the-air-beräkning (OAC) framstår som en genomförbar lösning, genom att tillåta summering av överförda signaler hos en gemensam mottagare genom analog amplitudmodulering. Utformad för att möjliggöra samtidig datainsamling och beräkning vid nätverkskanten, strävar OAC efter att minska den centrala systembelastningen, minska latens och energiförbrukning samtidigt som det möjliggör realtidsanalys. Denna metod är särskilt fördelaktig för federerad inlärning, en maskininlärningsteknik som fungerar över decentraliserade enheter. Dock medför OAC:s beroende av analog kommunikation märkbara utmaningar, inklusive signal distortion under överföring och begränsad tillgänglighet av enheter som stöder analoga moduleringar. Digital modulering är ett föredraget alternativ, erkänt för dess utmärkta kanalkorrigeringsegenskaper och bred acceptans i moderna trådlösa enheter. Trots detta uppfattas dess integration i OAC som ett betydande hinder, med överlappande digitalt modulerade signaler som hotar den grundläggande konceptet med samtidig datainsamling och beräkning.

Den första delen av avhandlingen ger en översikt över kommunikationssystem, med särskilt fokus på relevanta OAC-metodiker för analoga och digitala delar och deras tillämpning i ML, särskilt vid träning av federerade inlärningsmodeller. Därefter genomförs en omfattande litteraturöversikt angående analoga OAC-tekniker, där befintliga begränsningar inom detta område identifieras. Forskningens centrala drivkraft introduceras sedan, med förslag på en innovativ digital OAC-metod tillsammans med ett nytt perspektiv på kommunikationssystemmodeller utformade för att utföra beräkningen. Kapitlet avslutas med en sammanfattning av de huvudsakliga bidragen från varje artikel inkluderad i avhandlingen.

I den andra delen introducerar vi ChannelComp, en ny och banbrytande beräkningsmetod som är kompatibel med nuvarande digitala kommunikationssystem, inklusive smartphones och IoT-enheter. En detaljerad analys av ChannelComp:s funktioner avslöjar hur den möjliggör digitala moduleringsscheman för att utföra beräkningar, vilket adresserar en kritisk lucka i tidigare forskning. Dessutom möjliggör introduktionen av förkodare utformade för funktionsberäkning över den fleraccessa kanalen, kombinerat med ett ramverk för genomförbarhetsoptimeringsproblem, en sömlös integration med nuvarande system. Jämfört med OAC, begränsad till analoga moduleringar, uppvisar ChannelComp bredare beräkningsmöjligheter och efterlevnad av strikta beräkningstidsbegränsningar, vilket visar dess robusta potential för framtida massiva maskintypkommunikationer. Denna innovativa metod signalerar en lovande riktning mot hållbar och effektiv framtida trådlös kommunikation.

Acknowledgements

First and foremost, I am sincerely grateful to my supervisor, Professor Carlo Fischione, for the great help and support he has given me during these years. Moreover, I would like to thank my second supervisor, Dr. José Mairton Barros Da Silva Júnior, who has been so helpful and kind to me in the past two years. I have learned from my supervisors, and I am deeply thankful for the significant opportunity and honor they have provided me to work with them.

I extend my gratitude to my esteemed colleague at NSE and my remarkable friends for their contributions and camaraderie. My thanks go to Mohammad Azimi, Henrik Hellström, Jaume Angrea Perism, Giuseppe Nebbione, Viktor Engström, Oscar Stenhammar, Yang You, Emre Süren, Kiarash Kazari, Afsaneh Mahmoudi, Ezzeldin Zaki.

I also sincerely appreciate my esteemed friends within the WASP community. My gratitude goes to Alireza Dadras, Amandine Caut, Freidoon Zanganeeh, Hoomaan Maskan, Mehrdad Farahani, and Shivam Mehta.

Finally, I would like to express my appreciation to my family for their love and unconditional support throughout my life and my studies.

Saeed Razavi,
Stockholm, October 2023

Contents

List of Figures

List of Tables

Part I

Thesis Overview

Chapter 1

Thesis Formulation

As Internet of Things (IoT) continues to advance, it is set to transform lifestyles by enabling ubiquitous connectivity across a myriad of devices, marking a significant shift in wireless communications from human-centric to machine-centric paradigms. By 2025, IoT devices are projected to number 75 billion, vastly outnumbering mobile phone users, signaling a profound evolution in global connectivity [45]. In parallel, the emergence of 6G networks is anticipated to catalyze a new era of IoT applications, increasingly underpinned by machine learning (ML) technologies. These ML-driven applications, ranging from healthcare to smart cities and industrial automation, will necessitate the seamless collection, transmission, and processing of immense data volumes from numerous devices, forming an intricate and expansive network [22,37]. However, this proliferation of connectivity presents substantial challenges. To accommodate the anticipated data traffic, radio and computational resources must be significantly scaled up, posing a risk of overwhelming the capacity of existing communication systems.

Over-the-air computation (OAC) is emerging as a promising solution in response to these challenges by leveraging the superposition of electromagnetic waves. OAC is innovatively designed to enable simultaneous data collection and computation directly using *analog modulation* at the network's edge. This approach is strategically positioned to alleviate central system strain, significantly reducing latency and energy consumption while enabling real-time analytics and decision-making [7,21,44]. For instance, federated learning (FL), a form of ML that trains models across multiple decentralized devices while preserving data privacy, could greatly benefit from this approach. In summary, as IoT devices become an integral part of daily life and networks evolve towards machine-centric communication, innovations like OAC are poised to play a critical role in shaping a sustainable and efficient future for wireless communication. Further applications of OAC are shown in Figure 1.1.

Although OAC represents a promising concept for data aggregation in communication systems, such as federated edge learning, its reliance on analog communication presents significant challenges for ensuring reliable communications. This

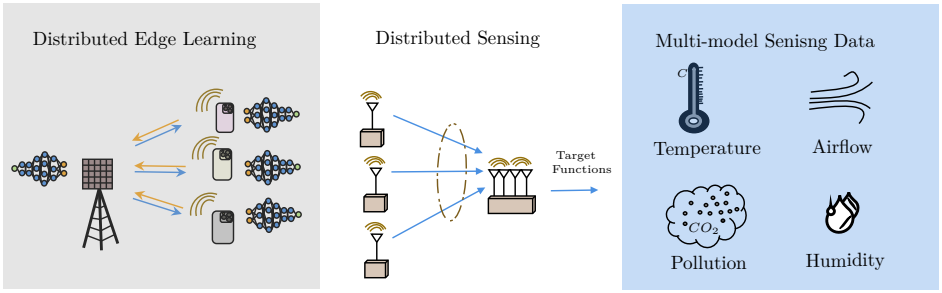


Figure 1.1: Illustration on OAC Applications. Using OAC in distributed edge learning ensures privacy and low-latency communications. A fusion center on an unmanned aerial vehicle or ground vehicle can monitor environments and collect sensing data. Given the brief contact with sensors and possible high sensor density, the standard communication methods may not satisfy low-latency requirements.

is due, in part, to the complexities introduced by channel ramifications, which can distort the integrity of analog signals during transmission [29]. Moreover, implementing OAC necessitates analog hardware systems that can utilize analog modulations. This becomes a limitation because of the limited number of wireless devices equipped to support analog modulations.

In contrast, digital modulation emerges as a potentially more advantageous alternative. Digital modulation is celebrated for superior channel correction properties and advanced source and channel coding capabilities. Furthermore, digital modulation technologies have seen widespread adoption across a broad range of contemporary wireless devices, attesting to their practical viability. However, adopting digital modulation in the context of OAC is widely regarded as a formidable challenge. This is because overlapping digitally modulated signals—unlike their analog counterparts—often generate generally meaningless or incomprehensible signals for function computation. This phenomenon undermines the core premise of OAC, which is to enable the simultaneous collection and computation of data directly at the network's edge [14, 40, 44].

This thesis investigates how to effectively implement digital modulation in the OAC domain. In the rapidly changing field of wireless communications, this research seeks to explore new possibilities of using OAC in digital modulation. The goal is to address important issues in how communication and computation resources are used, aiming to improve speed and reliability and ensure computation in a practical manner. The primary objectives of the thesis are encapsulated in the following research questions:

[RQ1] What constitutes the core challenges in utilizing digital modulations for computing, and what strategies can be developed to overcome these challenges?

[RQ2] Considering specific functions f, which constellation diagram is needed to

enable the computation of function f using digital OAC? Conversely, what are the corresponding implications of the computation for given modulations?

[RQ3] Given a particular modulation, is there a pathway to modify existing digital modulation schemes to execute computations for a general function f? If so, what are the constraints and boundaries about the class of functions and the nature of modifications?

[RQ4] Presuming a solution is available for executing computations across a broad category of functions, can it be characterized by low complexity and ease of implementation?

[RQ5] To what extent do channel noise and stochastic properties of the wireless channels influence the efficacy of this method? Does it exhibit robustness under these conditions?

[RQ6] Is there a mechanism to mitigate the channel's impact on executing the OAC, particularly when nodes have negligible or minimal information regarding the channel states?

These questions set the stage for an in-depth exploration to unearth novel insights and solutions. The thesis introduces a method named *ChannelComp*, which serves as a critical component in answering these research inquiries, thus helping shape the future direction of wireless communications.

Chapter 2

Theoretical Background

This chapter outlines the theoretical background that supports the thesis. In Section 2.1, the standard communication system is presented, including general signal models in a linear time-varying (LTV) channel, and both analog and digital modulations in standard communication systems are reviewed. Next, in Section 2.2, the OAC problem using analog modulation is explored. Section 2.3 goes over the basic concepts of machine learning and the role of the gradient descent algorithm in training. In Section 2.3, a distributed version of stochastic gradient descent and the FL problem are discussed, as well as how OAC can be used in FL to reduce communication and computation costs.

2.1 Communication Model

This section derives the multipath channels' input/output model. We demonstrate that multipath effects fit within a LTV system. After sampling the continuous-time channel for a simple point-to-point communication system, We develop a baseband representation that transitions into a discrete-time model. The model is further enhanced by incorporating additive noise. Indeed, transmitting a pulse over a multipath channel results in a received signal sequence, each component corresponding to either line-of-sight or distinct multipath scatterers [11]. A pronounced delay spread in the channel can significantly distort the received signal. Minimal delay spread ensures limited time spreading, while extensive spread induces notable distortion. The multipath channel's variability arises from transmitter or receiver mobility, causing shifts in reflector positions and multipath patterns. Although transmitting from a moving source produces variable signal components, these alterations span a longer timeframe than fading from static scatterers. Our study progresses from a general time-varying response to a more focused analysis of narrowband fading.

LTV channels are primarily characterized by multipath propagation and the Doppler effect. For an LTV channel $\mathbf{H}$ with P discrete paths, each path p is associated with a unique time delay τ_p and a Doppler shift ν_p. Let the transmitted

signal be $s(t)$, then the received signal $v(t)$ can be represented as [18]:

$$v(t) = \sum_{p=1}^{P} h_p s(t-\tau_p) \mathrm{e}^{j2\pi\nu_p t} + z(t), \tag{2.1}$$

where h_p, τ_p, and ν_p symbolize the complex attenuation factor, time delay, and Doppler frequency of the p-th path, respectively. Also, $z(t)$ is an additive white Gaussian noise (AWGN) process that is distributed according to $\mathcal{N}(0, \sigma)$. The expression in (2.1) characterizes the impact of P discrete specular scatterers (ideal point scatterers). Neglecting the noise term, this expression can be generalized to a continuum of scatterers as [13, 33]

$$v(t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S_{\mathbf{H}}(\tau, \nu) s(t-\tau) \mathrm{e}^{j2\pi\nu t} \mathrm{d}\tau \mathrm{d}\nu, \tag{2.2}$$

where the weight function $S_{\mathbf{H}}(\tau, \nu)$ is denoted as the (delay-Doppler) spreading function for the LTV channel $\mathbf{H}$. This function provides insight into the spread of the transmitted signal across time and frequency axes. At a specific delay-Doppler coordinate (τ, ν), the value of $S_{\mathbf{H}}(\tau, \nu)$ encapsulates the total complex attenuation and reflectivity of scatterers related to paths with delay τ and Doppler ν. This function also details the influence of the modified version $s(t-\tau)\mathrm{e}^{j2\pi\nu t}$ of the transmitted signal on the resultant signal $v(t)$. Note that (2.1) can be considered as a special case for (2.2) by choosing the following spreading function.

$$S_{\mathbf{H}}(\tau, \nu) = \sum_{p=1}^{P} h_p \delta(t-\tau_p) \delta(\nu-\nu_p), \tag{2.3}$$

where $\delta(\cdot)$ denotes Dirac delta function. For a narrow-band fading model, the multipath components are *non-resolvable*, which are combined into a single multipath component with delay $\tau \approx \tau_1 \approx \cdots \approx \tau_P$, an amplitude and phase corresponding to the sum of the different components. The amplitude of this summed signal will typically undergo fast variations due to the constructive and destructive combining of the non-resolvable multipath components [11]. Therefore, by neglecting the effect of the Doppler shift, for a narrow-band signal $s(t)$ with non-moving targets, the received signal $v(t)$ over the fading channel becomes

$$v(t) = hs(t-\tau) + z(t), \tag{2.4}$$

where $h \in \mathbb{C}$ indicates the path loss effects and $\tau \in [0, T_{\max}]$ models the delay created by the channel, wherein $T_{\max}$ shows the maximum possible delay originated by the channel.

2.1.1 Multiple Access Channel

If multiple devices communicate simultaneously over the same frequency band, the baseband representation of the received signal can be modeled using the MAC.

Consider a network with K nodes that can reach a server over a shared wireless channel. Let us denote the fading coefficient $h_k \in \mathbb{C}$ for node k and the transmitted signal by the same node as $s_k(t)$. Then, if all the nodes transmit over the same frequency band, the MAC models the received signal at the server as

$$v(t) = \sum_{k=1}^{K} h_k s_k(t - \tau_k) + z(t). \tag{2.5}$$

Ignoring the noise, the server obtains the weighted sum of the transmitted signal if all the nodes simultaneously and synchronously transmit their signal or $\tau_1 = \cdots = \tau_k$, i.e.,

$$v(t) = \sum_{k=1}^{K} h_k s_k(t). \tag{2.6}$$

Here, $s_k(t)$ is a band-pass signal with bandwidth B_k around its carrier frequency f_k. To communicate the information with the server, node k modulates its information massage x_k with signal $s_k(t)$ for transmitting over the MAC. In standard communication systems, we aim to recover the information transmitted by the nodes from the received signal $v(t)$. In the next section, we briefly overview the modulation procedure.

2.1.2 Digital and Analog Modulations

The transmission media serve as filters, differentially attenuating various frequencies. Realigning a signal's spectrum to frequencies less prone to significant attenuation is often advantageous to combat this. Certain media can notably distort digital signals, prompting the need for analog transmission using sinusoidal carriers. Furthermore, utilizing higher frequencies can enable smaller antenna and transceiver designs. When processing the signal, many circuits are optimized for specific frequency ranges. When these circuits handle diverse frequency bands, shifting the spectrum to an intermediate or optimal frequency is a good strategy. In all these situations, the process of modulation becomes an indispensable tool. Indeed, the modulation process alters the frequency and bandwidth of the message signal by utilizing a carrier signal capable of transmitting at a rate substantially higher than the baseband message signal.

In analog modulation, the characteristics of the modulated sinusoid (such as amplitude, frequency, or phase) can take a continuum of values depending on the source of information. The two common forms of analog modulation are amplitude modulation (AM) and frequency modulation (FM), a specific form of more general angle modulation. In particular, let $x_k \in \mathbb{R}$ be the massage of the node k, the AM modulated signal $s_k(t)$ is given by

$$s_k(t) = x_k \cos(2\pi f_c t + \phi_c), \quad \phi_c \in [0, 2\pi) \tag{2.7}$$

where f_c and ϕ_c show the frequency and phase of the carrier signal, respectively. In analog modulation, since the input value is taken from a continuum of values, the received signal is always associated with certain error levels depending on the channel noise variance.

In the case of digital signals, the information is in the form of a finite set of discrete symbols, which are called an alphabet. Indeed, x_k belongs to a finite set $\mathbb{F}_q$ with a maximum of q possible values. In this scenario, we use digital modulation for communication. Considering a binary alphabet, we have two distinct symbols: 0 and 1, with information being a sequence of these symbols. A binary digital signal depicts the "zero symbol" using a distinct waveform with a duration of T_s seconds. Similarly, the "one symbol" is symbolized by another waveform, generally of the same T_s duration. Consequently, transmission occurs at one bit every T_s second.

Given that each message, selected from a finite set, corresponds to specific amplitude, frequency, or phase values, we can deduce a discrete set of carriers linked to specific attributes from an alphabet. Digital modulations are classified based on the attribute to which the message is mapped. Specifically:

- Phase reversal keying or binary phase-shift keying (BPSK).
- Amplitude mapping results in amplitude-shift keying (ASK).
- Frequency mapping yields frequency-shift keying (FSK).
- Phase mapping produces phase modulation (PM).
- Amplitude and phase mapping combination denotes the quadrature amplitude modulation (QAM).

In particular, when utilizing BPSK modulation to communicate the value of the k-th node across an AWGN channel, the scalar $x_k \in \mathbb{R}$ is initially quantized to q bits and subsequently modulated with a carrier. In the context of BPSK, the scalar x_k undergoes one-bit quantization to $\tilde{x}_k \in \{0, 1\}$, leading to the relationship $\tilde{x}_k = x_k + q_k$, where q_k symbolizes the quantization error for the k-th symbol. This symbol is modulated using the basis signal s_k from Eq. (2.7) as

$$\mathscr{E}_k(\tilde{x}_k) = \begin{cases} E\cos(2\pi f_c t + \pi), & \tilde{x}_k = 1, \\ E\cos(2\pi f_c t), & \tilde{x}_k = 0. \end{cases} \tag{2.8}$$

The bandpass notation allows us to represent a signal by only its amplitude and phase. We expand a general form of the cosine function in (2.8) as follows:

$$\begin{aligned} s_k(t) &= E_k\cos(2\pi f_c t + \varphi_k), \\ &= \underbrace{E_k\cos(\varphi_k)}_{E_{1,k}}\cos(2\pi f_c t) + \underbrace{E_k\sin(\varphi_k)}_{E_{2,k}}\sin(2\pi f_c t), \\ &= E_{1,k}\cos(2\pi f_c t) + E_{2,k}\sin(2\pi f_c t). \end{aligned} \tag{2.9}$$

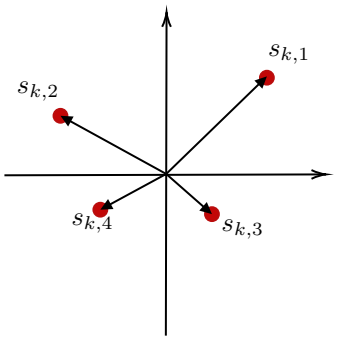


Figure 2.1: Example of constellation diagram of node k with its correspondence modulation vector $\boldsymbol{s}_k = [s_{k,1}, s_{k,2}, s_{k,3}, s_{k,4}]^\mathsf{T}$.

The given formula portrays a band-pass signal $s_k(t)$, in which both $E_{1,k}$ and $E_{2,k}$ function as its quadratic components. In the signal space, $s_k(t)$ is analogous to a complex number $s_k \in \mathbb{C}$, with $E_{1,k}$ and $E_{2,k}$ defining its real and imaginary segments, respectively. This complex value, s_k, can be viewed in the polar domain characterized by amplitude E_k and phase φ_k, representing a single-bit data. The k-th user transmitting q_k over a channel implies that s_k encompasses q_k complex values. Hence, the modulated signal, $s_k(t)$, can be depicted as a complex vector $\boldsymbol{s}_k \in \mathbb{C}^{q_k \times 1}$, where $s_{k,i}$ denotes its i-th element (refer to Figure 2.1). Additionally, the inner product of two complex signals, s_k and s_ℓ, within interval $[a, b]$ is defined [11]:

$$\langle s_k(t), s_\ell(t) \rangle = \int_a^b s_k(t) s_\ell^*(t)\, \mathrm{d}t. \tag{2.10}$$

Subsequently, two signals are defined as not linearly independent unless their inner product is zero. Moreover, the norm of signal s_k can also be defined in terms of its inner product as:

$$\|s_k(t)\| = \Big(\int_a^b |s_k(t)|^2 \mathrm{d}t \Big)^{1/2}. \tag{2.11}$$

In later sections, these notations will be important for analyzing the orthogonality and independence of complex signals.

2.2 Over-the-air Computation

OAC's principal notion capitalizes on the wireless channel's superposition attribute for computing the requisite function [10]. Due to the superposition's desirability, OAC accentuates interference, compelling all devices to transmit concurrently over identical radio resources, thus realizing non-orthogonal communication. Contrasting with conventional orthogonal communication systems, OAC's potential

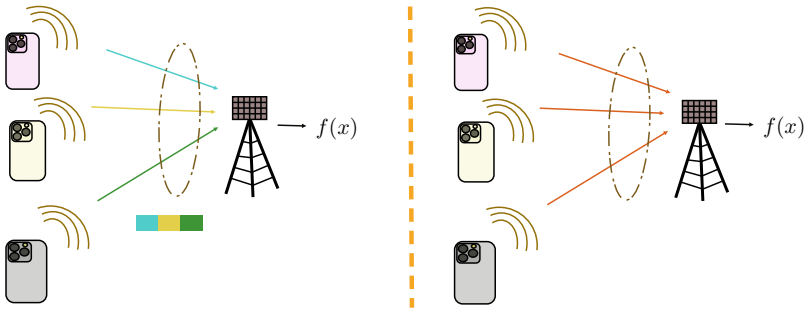


Figure 2.2: When contrasting the radio resource allocation in TDMA and FDMA with AirComp in a wireless setup with K devices, AirComp's key advantage emerges. Due to its comprehensive radio resource sharing capability, AirComp boasts a spectrum utilization improved by K. This enhancement can diminish bandwidth usage, reduce communication duration, or achieve synergy.

throughput enhancement is nearly directly proportional to the device count [21], yielding significant performance boosts in real-world applications.

2.2.1 AirComp: Analog Over-the-air

Analog OAC (AirComp) considers applications where a function of individual messages is desired rather than the individual messages themselves. Consider a communication network with a computation point (CP) and K nodes. The CP aims to compute a desired function $f(x_1, x_2, \ldots, x_K)$ whose input $x_k \in \mathbb{F}_q$ is a value owned at the k-th node. We note that specific functions can be computed through standard communication methods. Typically, nodes convey their messages x_k by allocating orthogonal radio resources distinctively for each node using a multiple access strategy, e.g., a straightforward instance being frequency division multiple access (FDMA). In FDMA, each node gets a unique frequency band, allowing for sequential message transmission. Transmissions are largely devoid of interference through orthogonal allocation, facilitating x_k message reconstruction at the receiver end. Once acquired, the receiver internally processes the desired function $f(\boldsymbol{x})$ where $\boldsymbol{x} := [x_1, \ldots, x_K]^\mathsf{T} \in \mathbb{R}^K$ is the vector of the messages x_ks.

For function computation, using orthogonal communication might be seen as overkill. Since individual messages x_k are discarded post $f(\boldsymbol{x})$ computation, reserving orthogonal radio resources solely for their reconstruction seems pointless. Rather, the aim is to compute the desired function leveraging the inherent summing capability of the wireless channel, as expressed by Eq. (2.6). To illustrate, consider the networks operating over an ideal MAC where $h_k = 1$ for all $k \in [K]$ and $z = 0$. Then, using an AM narrow-band modulated signal $s_k(t)$ for node k, Eq. (2.6) at

time slot T, becomes

$$v(T) = \sum_{k=1}^{K} x_k E_c \cos(2\pi f_c T + \phi_c) = \alpha_c \sum_{k=1}^{K} x_k, \tag{2.12}$$

where $\alpha_c := E_c \cos(2\pi f_c T + \phi_c)$. For any $\alpha_c \neq 0$, the received signal v gives us the summation function, i.e., $f(\boldsymbol{x}) = \sum_{k=1}^{K} x_k$ by dividing v with constant α_c. Without loss of generality and for the sake of simplicity, let $\alpha_c = 1$. This approach mirrors analog amplitude modulation, with devices encoding their messages g_k through signal power. Consequently, a single block of radio resources conveys the sum, in contrast to the K blocks needed for orthogonal communications. This efficiency can lead to bandwidth savings and decreased latency, as depicted in Figure 2.2.

Beyond just summing, various functions can be computed using tailored pre- and post-processing functions. Pre-processing functions $\varphi_k : \mathbb{R} \mapsto \mathbb{C}$ transform the messages into transmitted values, given by $s_k = \varphi_k(x_k)$. Subsequently, the post-processing $\psi : \mathbb{C} \mapsto \mathbb{R}$ is applied to the received value v to yield the target function as:

$$v = \psi(\sum_{k=1}^{K} \varphi_k(x_k)). \tag{2.13}$$

Using the AirComp formulation, we can compute a broad spectrum of functions that belong to the class of nomographic functions. A function is termed a *nomographic* function if it adheres to the following definition:

Definition 1 (Nomographic Function [31])**.** Given a compact metric space $\mathbb{S}^K$ for $K \geqslant 2$, a function $f : \mathbb{S}^K \mapsto \mathbb{R}$ is nomographic if there exist pre-processing functions $\varphi_k : \mathbb{S} \mapsto \mathbb{R}$ for each $k \in [K]$ and a post-processing function $\psi : \mathbb{R} \mapsto \mathbb{R}$, so that:

$$f(\boldsymbol{x}) = \psi\left(\sum_{k=1}^{K} \varphi_k(x_k)\right). \tag{2.14}$$

The compactness condition in Definition 1 ensures the commutability of continuous functions. The notation $\mathcal{F}(\mathbb{E})$ represents the set of all functions $f : \mathbb{E} \to \mathbb{R}$ within a closed unit interval $\mathbb{E} \subset \mathbb{R}$. Additionally, $\mathcal{C}^0(\mathbb{E})$ signifies the set of continuous functions that map to real values and have $\mathbb{E}$ as their domain. It is pertinent to note that $\mathcal{N}$ and $\mathcal{N}^0(\mathbb{E}^K)$ define the space of nomographic functions and the space of nomographic functions constrained by the continuity of both pre- and post-processing functions, respectively. We note two crucial theorems from Sprecher and Buck:

Theorem 1 (Sprecher'65 [35])**.** *Any function in $\mathcal{C}^0(\mathbb{E}^K)$ can be depicted using real, monotonic increasing pre-processing functions coupled with potentially discontinuous post-processing.*

Table 2.1: Examples for nomographic functions

Case	$f(x_1, \ldots, x_K)$	$\varphi_k(x_k)$	$\psi(x)$
Arithmetic Mean	$\frac{\sum_k^K x_k}{K}$	x_k	$\frac{x}{K}$
Geometric Mean	$(\prod_k^K x_k)^{\frac{1}{K}}$	$\log(x_k + \frac{1}{p_0(\epsilon)})$	$\exp(x/k)$
Weighted Sum	$\sum_k^K w_k x_k$	$w_k x_k$	x
p-norm	$(\sum_k^K x_k^p)^{\frac{1}{p}}$	$(x_k)^p$	$x^{\frac{1}{p}}$
Maximum	$\max_k x_k$	$x_k^{p_0(\epsilon)}$	$x^{\frac{1}{p_0(\epsilon)}}$
Total Number of Nodes	K	1	x

Theorem 2 (Buck'79 [5])**.** *Every function in* $\mathcal{F}(\mathbb{E}^K)$ *is nomographic, implying* $\mathcal{N}(\mathbb{E}^K) = \mathcal{F}(\mathbb{E}^K)$*.*

However, if continuous pre- and post-processing functions are sought for an arbitrary function f, Theorem 2 does not apply. Instead, one might approximate a desired function with nomographic functions. An example under Definition 1 is the geometric mean on $\mathbb{E}^K$, which can be approximated with $\varphi_k(x) = \ln(x + 1/p_0(\epsilon))$ and $\psi(x) = \exp(x/K)$ where $p_0(x) > 0$. For a detailed overview of nomographic functions and associated properties, we recommend [8, 31, 36]. A few examples of nomographic functions and their associated pre- and post-processing functions can be found in Table 2.1.

2.2.2 Orthogonal Coded Over-the-air

In this section, we study the methods employing variants of source and channel coding for the OAC problem. This approach was first introduced in [21], where the authors used the linearity properties of the nested lattice codes and computation over Gaussian channels in finite fields. Following nested lattice coding for Gaussian channels, in [9], introduces a digital OAC scheme to enhance the computation reliability of functions within a real-valued AWGN channel.

In particular, to calculate the sum function, i.e., $f = \sum_{k=1} s_k$, the process is initiated by quantizing the input symbol s_k and mapped into a positive integer $p_k \in \{0, 1, \ldots, q\}$ where q denotes the quantization level. Next, the integers p_k are converted to the base-α numeral system, where $\alpha := K(q-1)+1$. Indeed, for the n-th symbol, the input integer $p_k[n]$ is encoded into massage $g_k[n]$ as follows

$$g_k[n] := (p_k[n])_\alpha, \quad \alpha := K(q-1)+1. \tag{2.15}$$

The chosen value of α eliminates a potential carry digit for the superposed message among K nodes. Afterward, for a given orthogonal bandwidth B and N number of functions, a nested lattice code is constructed from $\mathbb{Z}_p^N$ to $\mathbb{R}^B$ for a prime $p \geqslant \alpha^N$,

where

$$\mathbb{Z}_p^N := \underbrace{\mathbb{Z}_p \times \cdots \times \mathbb{Z}_p}_{n \text{ times}}. \tag{2.16}$$

Using the generator matrix $\boldsymbol{G} \in \mathbb{R}^{B\times N}$ for any $B > N$, the N messages derived are used to compute the codeword $\boldsymbol{c}_k \in \mathbb{C}^B$. Mathematically, we have

$$\boldsymbol{c}_k = \boldsymbol{G}\boldsymbol{g}_k, \tag{2.17}$$

where $\boldsymbol{g}_k := [g_k[1], \ldots, g_k[N]]^\mathsf{T}$ is the vector of N encoded messages of node k.

Next, neglecting the noise of the channel, the CP receives the summation of the code-words, i.e.,

$$\boldsymbol{y} = \sum_{k=1}^{K} \boldsymbol{c}_k + \boldsymbol{z}. \tag{2.18}$$

Note that transmitting the N elements of the vector $\boldsymbol{c}_k$ occurs over B orthogonal resources. To decode the received data $\boldsymbol{y}$, the CP is processed through two main stages. First, an Euclidean nearest neighbor decoder is employed to obtain the superposed message $\boldsymbol{v} := \sum_k \boldsymbol{c}_k$ corresponding to the linearity of the code. Then, having the invertible generator matrix $\boldsymbol{G}$, we can write

$$\boldsymbol{g} := \boldsymbol{G}^{-1}\boldsymbol{v} = \boldsymbol{G}^{-1}\sum_{k=1}^{K} \boldsymbol{c}_k = \boldsymbol{G}^{-1}\boldsymbol{G}\sum_{k=1}^{K} \boldsymbol{g}_k = \sum_{k=1}^{K} \boldsymbol{g}_k. \tag{2.19}$$

Finally, the superposed message is expressed in base α, yielding the sum of the quantization outcomes, symbolized as $(\sum_k p_k[1] \sum_k p_k[2] \cdots \sum_k p_k[N])_\alpha$.

The strategy identified as orthogonal coded OAC leverages B orthogonal communication resources. Even though digital OAC methods consume a greater amount of resources, making them not as spectrally efficient compared to AirComp methods, they find their application predominantly in straightforward digital modulations, such as ASK. It is important to note that integrating more complex modulation techniques, QAM and other high-order modulations, into the digital OAC framework presents a significant challenge, not offering a straightforward pathway for coherent implementation. In fact, implementing OAC using standard digital communication results in a meaningless overlapping of the digital signals by the superposition property of the radio waves.

2.2.3 Related Works on Analog Over-the-air

The principal objective of this thesis is to propose novel computation methods for a fully digital OAC architecture, including an analysis of its advantages and limitations. This section provides a synopsis of existing methods and resolutions to the analog OAC computation methods with partial digital components, followed by

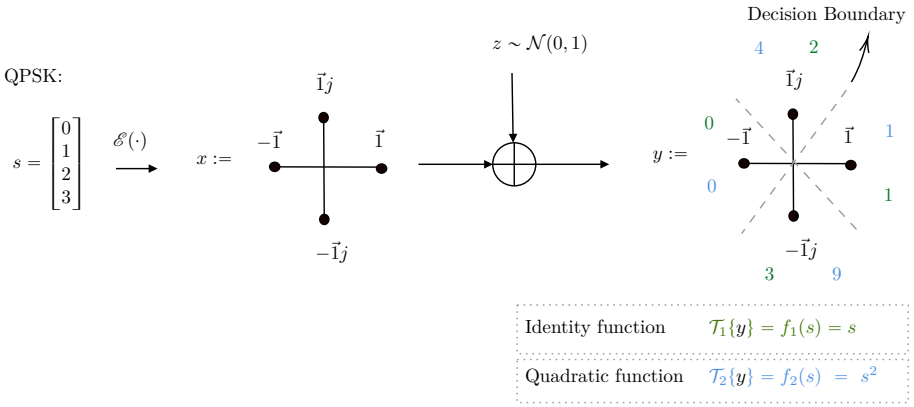


Figure 2.3: Digital modulation mechanism in standard point-to-point communication versus ChannelComp digital OAC. The function $\mathscr{E}(\cdot)$ encodes the real input data $s \in \mathbb{R}$ to the complex domain with $x \in \mathbb{C}$. Tabular $\mathcal{T}(\cdot)$ maps the complex value y to the desired function, which can be the identity function for standard communication.

a brief introduction to the problem's formalism. In network architectures including K nodes, digital modulation schemes are employed for computing the function $f(s_1, \ldots, s_K)$ over $s_1, \ldots, s_K$ input variables. This disputes the utilization of analog OAC. Note that the computational processes are constrained to only one shared communication resource. Consequently, all nodes synchronize in frequency, time slot, and coding schema for communication with the CP.

Specifically, each node K transmits its respective signal s_k through a digitally modulated signal x_k, generated by a digital encoder denoted as $\mathscr{E}_k$, for all k in the set $[K]$. Upon concurrent transmission from all nodes, the CP receives the resultant signal y, which can be expressed as:

$$y(t) = \sum_{k=1}^{K} x_k(t) + z(t) = \sum_{k=1}^{K} \mathscr{E}_k(s_k(t)) + z(t). \tag{2.20}$$

Here, $z(t)$ signifies an AWGN with $\mathcal{N}(0, \sigma_z)$. The ultimate objective is to compute the desired function f based on the received signal y. A pivotal aspect to note is the induction of a novel constellation diagram due to the aggregation of x_k signals. Specifically, Eq. (2.20) culminates in a unique constellation diagram for vector $\vec{y}$, contingent upon both the aggregate number of nodes K and the modulation schemes employed for each x_k. It merits attention that this emergent constellation diagram deviates from the original diagram associated with the individual transmitting nodes. The comparison between the digital modulation mechanism in standard point-to-point communication versus digital OAC is depicted in Figure 2.3 for case $K = 1$.

However, specialized instances of the function f have been investigated [43]. For example, [43] has presented strategies for digital aggregation, such as one-bit broadband digital aggregation (OBDA), restricted to BPSK or quadri-phase shift keying (QPSK). The target function in this work has aimed to resolve the signSGD optimization [2] and has formalized as $f = \text{sign}\left(\sum_k \text{sign}(s_k)\right)$. Subsequent adaptations of OBDA have been proposed that are focused on joint channel decoding and aggregation decoding mechanisms, specifically tailored for digital OAC. Moreover, non-coherent communication approaches for single and multi-cell configurations, employing pulse-position modulation and FSK, are explored in [1, 27, 28]. Nevertheless, the previously mentioned OBDA investigations are confined to specific function types or particular ML training algorithms. Additionally, alternative schemes [29] adopt multiple communication resources, such as time and frequency, for function computation, diverging from the principles of digital OAC. Lastly, [41] discusses a methodology for estimating weighted sums using binary components of symbols transformed into BPSK representations, emphasizing the relevance of linearity in the decomposition process for weighted sum estimation. The cornerstone observation is the locational shift of the retrieved symbol into a non-standard pulse-amplitude modulation (PAM) domain post-superposition.

The existing digital aggregation methods enforce analog AirComp over digital communication. Such enforcement works in very few and particular cases of digital modulations where it appears highly inefficient regarding communication resource usage, such as satellite communications [15], integrated communication and sensing [23], unmanned aerial vehicles [42], distributed consensus [6].

2.3 Machine Learning

In this section, we explore ML and its overarching objectives, delving into the mathematical formulation inherent in supervised learning problems and the optimization techniques employed in training ML algorithms. We commence by outlining the foundational principles of supervised ML methods, with a particular emphasis on optimization. This is followed by a discussion on key optimization strategies utilized in ML training, notably gradient descent (GD), stochastic gradient descent (SGD), and FL. The section concludes with an introduction to the application of OAC to federated edge learning (FEEL) problem.

At its core, ML is centered on empowering computers to learn from a designated set of data, referred to as *training data*. This involves training computers to discern essential features for predictive tasks and to draw inferences based on this data. ML algorithms leverage mathematical models constructed from available training data to generate predictions or decisions, enabling the systems to continually adapt and enhance performance through the iterative procedure and training of new data.

Let us define our problem by introducing some foundational terms and concepts:

- **Data Sample Vector ($\boldsymbol{x}$):** We define $\mathcal{X}$ as the collection of training data samples. Any given $\boldsymbol{x} \in \mathcal{X}$ represents the data sample vector with dimension

d. This can be expressed mathematically as $\boldsymbol{x} \in \mathbb{R}^d$.

- **Training Label (y):** $\mathcal{Y}$ is the labels corresponding to the data samples. Specifically, for any data sample $\boldsymbol{x}$, the corresponding label is denoted as $y \in \mathbb{R}$. A data sample and its label can be represented as the pair $(\boldsymbol{x}, y)$.
- **Hypothesis Function (h):** The learning algorithm outputs this function as $h : \mathcal{X} \to \mathcal{Y}$. Its primary function is to map an input from space $\mathcal{X}$ to output space $\mathcal{Y}$. The goal is to predict $y \in \mathcal{Y}$ for an arbitrary $\boldsymbol{x} \in \mathcal{X}$. The hypothesis functions can be linear functions mapping input feature vectors to real numbers. They also can be selected from nonconvex functions, such as neural networks. Our problem can be succinctly framed as the quest for selecting an appropriate function from a hypothesis set, defined as $\mathcal{H}$. This set is a subset of the exhaustive family of all potential functions.
- **Objective of the Learning Algorithm:** The main objective is to discern a hypothesis h most similar to a designated loss function. This loss function, represented as $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, serves as a metric to quantify the deviation between the proposed hypothesis, $h(\boldsymbol{x})$, and the actual output, y [20]. The loss functions are typically bounded, but these conditions do not always hold. Common examples of loss functions include the zero-one, ℓ_2, ℓ_1 norms, or binary cross-entropy, which are typically unbounded.

In our intent to understand the learning problem, let us cast it as an optimization problem and discuss its properties. Given our dataset, the learning problem is encapsulated by the optimization:

$$h^* = \arg\min_{h \in \mathcal{H}} \mathbb{E}_{x \sim \mathcal{X}} \big[\ell(h(\boldsymbol{x}), y) \big], \tag{2.21}$$

where $\mathcal{H}$ represents the set of potential hypothesis functions. However, addressing this optimization across a generic family of hypotheses is complicated. A widely accepted strategy to handle this complexity is to narrow the hypothesis search space to a specific functional form, expressed as $h(\cdot)$. The function, in turn, is driven by a real vector parameter, $\boldsymbol{w} \in \mathbb{R}^d$. In machine learning terminology, this vector is denoted as the ML parameter, integral to the parameterized hypothesis, denoted as $h(\boldsymbol{x}, \boldsymbol{w})$. The training's objective focuses on pinpointing the paramount value for the learning parameter $\boldsymbol{w}^*$. This is realized by attenuating a specific loss function $\ell(h(\boldsymbol{x}, \boldsymbol{w}), y)$, which gauges the gap between the model's forecasted output $h(\boldsymbol{x}, \boldsymbol{w})$ and the actual output y. Mathematically, the quest to obtain $\boldsymbol{w}^*$ is described by:

$$\boldsymbol{w}^* = \arg\min_{\boldsymbol{w}} \ell(h(\boldsymbol{x}, \boldsymbol{w}), y) = \arg\min_{\boldsymbol{w}} \frac{1}{N} \sum_{i=1}^{N} \ell(h(\boldsymbol{x}_i, \boldsymbol{w}), y_i). \tag{2.22}$$

The training data comprises a collection of tuples $(\boldsymbol{x}, y)$ defined as $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$. Here, the loss function $\ell(\cdot)$ is a deliberate choice to quantify the divergence between

predicted and factual outputs. The aim is to discover a hypothesis $h(\cdot)$ that reduces our learning error.

During training, error assessment revolves around solving Eq. (2.22) using N samples. The testing phase involves pitting the hypothesis function $h(\boldsymbol{x}_j, \boldsymbol{w})$ against the output $y_j \in \mathcal{Y}$ for untrained samples $\boldsymbol{x}_j \in \mathcal{X}$. Classification accuracy, a pivotal metric, demarcates the correct predictions' ratio to incorrect ones. This underpins the essence of selecting an apt hypothesis.

2.3.1 Gradient Descent

A commonly employed method to tackle the optimization problem presented in (2.22) using a dataset of $\mathcal{D} = \{(\boldsymbol{x}_j, y_j)\}_{j=1}^N$ is the GD algorithm. As a key optimization technique, GD is extensively utilized for minimizing various functions, including loss functions in ML models. In the realm of deep neural network (DNN) optimization, its foundational-most advanced ML libraries provide GD implementations through the backpropagation mechanism [26]. Essentially, the GD algorithm iteratively refines the model's parameters based on training data until its loss function (often referred to as the objective function) converges to a local minimum [25].

The GD algorithm initializes with an arbitrary selection of $\boldsymbol{w}_0$. In subsequent steps, it modifies the parameters in the opposite direction of the gradient of $\ell(h(\boldsymbol{x}_j, \boldsymbol{w}), y_j)$, which is expressed as $\boldsymbol{g}_j(\boldsymbol{w}) := \nabla \ell(h(\boldsymbol{x}_j, \boldsymbol{w}), y_j)$, for each $j \in [N]$. Here, N denotes the size of the dataset, or equivalently, $|\mathcal{D}|$. Thus, for every iteration $t \in [T]$, the GD update is defined as:

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} - \frac{\eta}{N} \sum_{j=1}^{N} \boldsymbol{g}_j(\boldsymbol{w}_{t-1}), \quad t \in [T], \tag{2.23}$$

where $\eta \in (0, 1)$ symbolizes the step size for the GD updates. Notably, the gradient computation involves all samples from the dataset $\mathcal{D}$. The ultimate goal of the GD algorithm is to minimize the loss function, $\ell(h(\boldsymbol{x}, \boldsymbol{w}), y)$, with respect to the model parameter $\boldsymbol{w} \in \mathbb{R}^d$, given the data $\mathcal{D}$. This standard GD approach is also called the full gradient descent.

Obviously, the convergence of the GD depends on the characteristics of the objective function f. Before discussing the convergence of the GD method, we review the basic definitions of optimization theory [4], such as convex, strongly convex, and L-smooth functions.

Definition 2. (**Convexity**) A set $\mathcal{C}$ is called convex if for any $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{C}$ and any $\alpha \in [0, 1]$, we have

$$\alpha \boldsymbol{x}_1 + (1 - \alpha)\boldsymbol{x}_2 \in \mathcal{C}. \tag{2.24}$$

Similarly, a function $f : \mathbb{R}^d \to \mathbb{R}$ is convex if **dom** f is a convex set and for all $\boldsymbol{x}_2, \boldsymbol{x}_2 \in \mathbb{R}^d$, and for any $\alpha \in [0, 1]$, we have

$$f(\alpha \boldsymbol{x}_1 + (1 - \alpha)\boldsymbol{x}_2) \leqslant \alpha f(\boldsymbol{x}_1) + (1 - \alpha) f(\boldsymbol{x}_2). \tag{2.25}$$

Definition 3. (**Smoothness**) A continuously differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ with the corresponding gradient $g(\boldsymbol{w})$ is L-smooth, $0 < L < \infty$, if for all $\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathbb{R}^d$, we have

$$f(\boldsymbol{w}_2) \leqslant f(\boldsymbol{w}_1) + g(\boldsymbol{w}_1)(\boldsymbol{w}_2 - \boldsymbol{w}_1)^\top + \frac{L}{2}\|\boldsymbol{w}_2 - \boldsymbol{w}_1\|_2^2. \tag{2.26}$$

Equivalently, we have

$$\|g(\boldsymbol{w}_1) - g(\boldsymbol{w}_2)\|_2 \leqslant L\|\boldsymbol{w}_1 - \boldsymbol{w}_2\|_2. \tag{2.27}$$

Note that in optimization theory, the GD method is recognized for its optimality when applied to the convex loss function, owing to its convergence to a global minimum. Conversely, its application to non-convex functions is often considered suboptimal, as it may converge to local minima.

2.3.2 Stochastic Gradient Descent (SGD)

For large-scale datasets where $|\mathcal{D}| \gg 100$, computing the gradient over the entire dataset creates huge computational challenges. In such scenarios, SGD emerges as a more computationally efficient alternative to the traditional GD. Instead of utilizing the full dataset, SGD computes the gradient using a subset, $\mathcal{D}_t \subset \mathcal{D}$, at each iteration t [32]. This approach greatly mitigates computational demands, especially with large ML training datasets.

More precisely, during the t-th iteration, a subset $\mathcal{D}_t$ of size N_t, where $N_t \ll N$, is sampled independent and identically distributed (iid) from the dataset. The gradient is computed based only on this subset instead of computing the full gradient $\nabla\ell(\boldsymbol{w})$. This gives rise to:

$$g_t(\boldsymbol{w}_t, \mathcal{D}_t) := \frac{1}{N_t}\sum_{j\in \mathrm{I}_t} \nabla\ell(h(\boldsymbol{x}_j, \boldsymbol{w}_{t-1}), y_i), \quad t \in [T], \tag{2.28}$$

with I_t denoting the randomly chosen set of indices from $[N]$ at iteration t. Notably, in a standard SGD setup, N_t is fixed and set to 1. Using this gradient, the update rule for SGD at each iteration becomes:

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} - \eta_t g_k(\boldsymbol{w}_t, \mathcal{D}_t), \quad t \in [T]. \tag{2.29}$$

With SGD, there is no longer a guarantee that the algorithm moves in a descent direction, and regardless of the choice of η_t, there are no deterministic convergence guarantees. Indeed, $g_k(\boldsymbol{w}_t, \mathcal{D}_t)$ can be considered an estimator of the true gradient $\sum_{j=1}^N \boldsymbol{g}_j/N$. When employing a fixed mini-batch size of $N_t = B \gg N$ alongside a diminishing step size η_t, it is possible to approximate the full gradient using an unbiased estimation:

$$\mathbb{E}_{\mathcal{D}_t}\{g_t(\boldsymbol{w}_t, \mathcal{D}_t)\} = \nabla\ell(\boldsymbol{w}_t). \tag{2.30}$$

With sufficiently many iterations, we expect SGD to move in the same direction as full GD, eventually converging to a region of optimality around a local minimum [3]. Given a Lipschitz smoothness condition on the gradient (Definition 3), the convergence rate of the mini-batch SGD is defined as $O(1/\sqrt{TN}+1/T)^3$ [38]. While SGD offers the advantage of reduced iteration cost and memory consumption, its convergence is often slower, and it may falter when encountering strong convexity.

The discussion thus far has focused on ML models operating on individual devices. As we transition to environments where IoT devices function within distributed systems, often in synergy with other devices and servers, a broader perspective is necessary. In the following subsection, we will delve into distributed ML, laying the groundwork for this thesis.

2.3.3 Federated Learning (FL)

The prevalent use of DNN has led to significant advancements using SGD variations for optimization [19]. With the vast data from IoT devices, distributed ML algorithms, especially federated learning (FL), have gained traction. FL offers data privacy by eliminating data sharing between devices and central servers [17, 19]. Moreover, FL considers the communication cost for the learning procedure more than distributed ML approaches [39]. The key characteristic of FL is that every network node runs multiple local training steps before communicating an update to the server. This reduces communication costs significantly, especially in large DNN with $d \gg 10$ million model parameters [34].

Let us elaborate on a specific instance within distributed computing, where we focus on a network comprising a CP and K nodes, as we previously discussed in Section 2.1.1. This scenario is an essential use case where we aim at training a universal model, denoted by $\boldsymbol{w}^*$, without sharing private data among different entities. Specifically, we allocate to each node k an individual local training dataset, symbolized by $\mathcal{D}_k$, where $|\mathcal{D}_k|$ quantifies the number of samples that fall under ED k. Moreover, we can associate the local loss function, utilized by ED k, as $L_k(\boldsymbol{w})$, where $\boldsymbol{w} \in \mathbb{R}^d$ signifies the model parameters, with d determining their size.

Now, let us break down the FL procedure into distinct steps:

1. **Initialization**: The process starts with the CP broadcasting an initial parameter, $\boldsymbol{w}_0$, to all the nodes in the network.

2. **Local Data Retention**: Each node, indexed by $k \in \{1, 2, \ldots, K\}$, holds a subset $\mathcal{D}_k = \{(\boldsymbol{x}_j^k, y_j^k)\}_{j=1}^{N_k}$ of the overarching dataset, $\mathcal{D}$. This complete dataset can be expressed as: $\mathcal{D} = \cup_{k=1}^{K} \mathcal{D}_k$.

3. **Local Iterations**: Using the parameter vector $\boldsymbol{w}_0$ received from the CP, every node performs multiple localized iterations of the GD, following the scheme detailed in [19]. This sequence leads to the derivation of its individu-

alized parameter $\boldsymbol{w}_t^k$ as:

$$\boldsymbol{w}_t^k = \boldsymbol{w}_{t-1} - \frac{\alpha_t}{N_k} \sum_{j=1}^{N_k} \nabla \ell(h(\boldsymbol{x}_j^k, \boldsymbol{w}_{t-1}), y_j^k), \quad t \in [T], \tag{2.31}$$

where α_t represents the step size.

4. **Device to Server Communication**: Subsequent to the local iterations, each device conveys its $\boldsymbol{w}_t^k$ to the server CP.

5. **Global Update**: The server compiles these inputs and updates $\boldsymbol{w}_t$ using a method known as a global update:

$$\boldsymbol{w}_t = \sum_{k=1}^{K} \rho_k \boldsymbol{w}_T^k, \quad t \in [E], \tag{2.32}$$

where ρ_k is the weighting factor defined as the ratio $\rho_k := N_k/K$.

6. **Reiteration and Convergence**: This process continues iteratively, fine-tuning the parameters until convergence.

Note that the update mechanism for $\boldsymbol{w}_t^k$ aligns with the global update strategy in FL as specified in (2.31). Among the myriad of methodologies within the FL domain, federated averaging (FedAvg) stands out. This approach differentiates itself from standard FL by implementing unique computations for local parameters. While FL presents the advantage of reduced computational complexity due to its single local iteration, FedAvg introduces multiple local iterations, which allow it to significantly reduce the requisite communication iterations for training by a factor of T. This contraction in communication rounds is far from trivial; it can translate into multiple orders of magnitude decrease in complexity, particularly when the mission is to train DNNs on disjointed data sources [19, 24].

One potential drawback of the FL scheme is the slowness of the connection between the central server and a device. Their communication can become slow, especially in large-scale networks with vast volumes of data. This latency is particularly important in time-sensitive applications, such as smart healthcare [16]. Consequently, there is a high demand to create communication-efficient FL strategies. Furthermore, given the computational limitations of devices, there has been a marked shift towards formulating computation or energy-efficient FL methods in recent years.

2.3.4 Over-the-Air Distributed Machine Learning

For FEEL, communication and computation can be regarded as distinct tasks. In each iteration, the CP must acquire the local model parameters (or gradients) from K nodes to compute d functions, which means dK parameters need to be

transferred in the uplink. When d is large, this task becomes computationally intensive; consequently, the latency grows linearly with the number of nodes when an orthogonal multiple access scheme is used.

In contrast, the cost with OAC equals that of a single node, as all nodes transmit simultaneously to compute d functions, such as the average of the local model parameters. Therefore, the training can be completed with higher communication efficiency and lower bandwidth when utilizing OAC.

Specifically, OAC allows approximating the arithmetic mean of local gradients, as represented in equation (2.5), during the uplink aggregation phase. OAC can achieve uplink aggregation of gradients by following methods explained in Section 2.2 regarding the computation of nomographic functions in (2.13). A simple OAC implementation may involve pre- and post-processing functions, such as $\varphi_k(\boldsymbol{w}_t^k) = \boldsymbol{w}_t^k h_k^*/|h_k|^2$ and $\psi(v) = v/K$. When nodes transmit simultaneously over the fading MAC, the CP receives:

$$v = \psi\Big(\sum_{k=1}^{K} h_k \varphi_k(\boldsymbol{w}_t^k) + z\Big) = \frac{1}{K}\sum_{k=1}^{K} \boldsymbol{w}_t^k + \frac{z}{K}. \tag{2.33}$$

However, due to channel noise, distortions and lack of synchronization in the global model's estimation may occur, thus affecting the convergence procedures and potentially making the FEEL algorithms converge slower than the error-free version. Additionally, in practical situations where nodes have limited transmission power, assuming they always have sufficient power to invert their channels may be unrealistic. This inability to fully compensate for the channel coefficients' effects results in estimated values with Rayleigh distribution [10]. In this thesis, we investigate two main challenges of FEEL with OAC: channel effects and synchronization. We specifically focus on situations where the network nodes are unaware of the channel state information or synchronization delays, a condition termed in the literature as *blind*. Therefore, we investigate the possible solutions for blind FEEL by using statistical signal processing techniques and communication coding tools.

Chapter 3

Research Contributions

In this thesis, we challenge existing norms in wireless communications and introduce a groundbreaking computing approach, termed *ChannelComp*. This innovative method is engineered to be fully compatible with current digital communication systems, such as those in smartphones and IoT devices. We conduct a detailed analysis of the functions ChannelComp can compute and explore how it enables digital modulation schemes to perform computations, addressing a significant gap in prior work. We also formulate a feasibility optimization problem, a mathematical framework that identifies the optimal modulation scheme for over-the-air computation of arbitrary functions. We introduce pre-coders that adapt existing digital modulation schemes for function computation over the MAC, allowing seamless integration with current systems. Compared to the OAC method, restricted to analog modulations, ChannelComp exhibits broader computational capabilities. Moreover, it is designed to manage numerous devices simultaneously, adhering to stringent computation time constraints, showcasing its potential as a robust solution for future massive.

This thesis investigates the analysis of the digital over-the-air computation over the MAC to address the primary research questions. The chapters presented in the second part of this thesis are based on the following published or submitted manuscripts:

[C1] **S. Razavikia**, J. M. B. da Silva Jr, C. Fischione, "Computing Functions Over-the-Air Using Digital Modulations," *IEEE International Conference on Communications*, Rome, Italy, 2023.

[J1] **S. Razavikia**, J. M. B. da Silva Jr, C. Fischione, "Channel Computing: Computation by Communications," to appear in *IEEE Transactions on Communications*, 2023.

[P1] **S. Razavikia**, C. Fischione "Channel Computing: Computation by Communications," (European Patent) filed June 2022.

Table 3.1: Research questions and paper contributions

Research Question	[J1]	[J2]	[J3]	[C1]	[C2]	[P1]
RQ1	✓	✓	✓	✓	-	✓
RQ2	✓	✓	-	✓	-	✓
RQ3	✓	-	-	-	-	✓
RQ4	-	-	✓	-	-	✓
RQ5	✓	-	✓	-	✓	✓
RQ6	-	-	✓	-	✓	✓

[J2] **S. Razavikia**, J. M. B. da Silva Jr, C. Fischione, "SumComp Coding: Digital Over-the-Air Computation via the Ring of Integers," submitted to *IEEE Transactions on Communications*.

[J3] **S. Razavikia**, J. M. B. da Silva Jr, C. Fischione, "Blind Federated Learning via Over-the-Air q-QAM," submitted to *IEEE Transactions on Wireless Communications*.

[C2] **S. Razavikia**, J. A. Peris, J. M. B. da Silva Jr, C. Fischione,"Blind Asynchronous Over-the-Air Federated Edge Learning," *IEEE Globecom Workshops*, Rio de Janeiro, Brazil, 2022, pp. 1834-1839.

In Table 3.1, we present the contribution of each paper towards addressing the main research questions of this thesis. Below, we briefly present the main contributions of the publications, which are included in separate chapters.

3.1 Channel Computing: Computation by Communications

This topic is investigated in Chapter 4 and is based on the material provided in [C1], [J1], and [P1]. Indeed, we identify the main challenges for employing digital modulation in OAC method. There was a common belief that digital modulations are generally unfeasible for computation tasks because the overlapping of digitally modulated signals returns signals that seem to be meaningless for these tasks. Chapter 4 breaks through this belief to address this issue and proposes a fundamentally new computing method, ChannelComp, for performing over-the-air computations by any digital modulation.

In particular, we present a novel digital channel computation method for computing functions over MAC by any digital modulation format and for any finite value functions. Specifically, we consider the problem of computing a K-variate function $f(x_1, \ldots, x_K) : \mathbb{R}^K \mapsto \mathbb{R}$, where $x_k \in \mathbb{R}$ for $k = 1, \ldots, K$, belongs to node k of a network with K nodes. The nodes use digital modulations to transmit the values x_k over a MAC to a server that needs to compute the function f of the nodes' values. We establish the conditions for computing general functions. For a given function, these conditions lead to optimization problems whose solutions

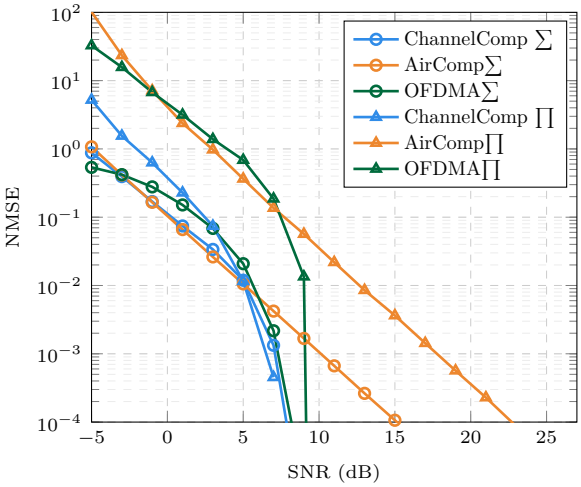


Figure 3.1: Performance comparison between our proposed ChannelComp, the traditional AirComp, and OFDMA methods in terms of NMSE error averaged over $N_s = 100$ when values of the function to be computed are originally quantized in [J1]. The input values are $x_k = \{0, 1, 2, \ldots, 7\}$ and the desired functions are $f_1 = \sum_{k=1}^{4} x_k$ and $f_2 = \prod_{k=1}^{4} x_k$.

determine the parameters of the digital modulation, resulting in a correct computation over-the-air. Specifically, we propose a feasibility optimization problem to obtain the parameters of the used digital modulation. Such a feasibility problem is NP-hard, and to overcome such complexity, we develop a convex relaxation that can be solved using traditional solvers, such as CVX [12].

ChannelComp can compute functions for a finite number of its input domains, which is typical of digital communication systems because they only handle quantized values. In the numerical experiments presented in [J1], which is reproduced here in Figure 3.1 for simplicity, ChannelComp outperforms OAC regarding computation error for various important functions while consuming the same communication resources. For example, for computing the product function, ChannelComp obtains a 10 dB performance improvement compared to OAC regarding the normalized mean square error. This result is remarkable because ChannelComp does not need to use analog modulations and only relies on well-known digital modulations.

Moreover, we show how to adapt existing digital or analog modulation schemes using pre-coders to compute the desired function over the MAC. Note that the rationale behind ChannelComp draws from the benefits of digital modulations, yet its scope can also be extended to include analog modulations. One of the key benefits of ChannelComp is its innate compatibility with existing digital systems. By contrast, if we wish to implement OAC on existing digital systems, it is necessary to enforce analog communication systems to work on top of current digital systems,

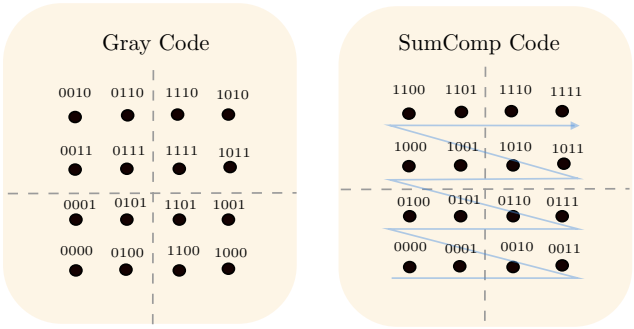


Figure 3.2: Gray code vs SumComp code for QAM 16 modulation. In the SumComp algorithm, each row exhibits a uniform increment in the value of symbols, a characteristic not shared with the Gray code scheme. This attribute facilitates the accurate computation of summations derived from the constellation points induced over the MAC. Conversely, employing the Gray code for node constellation engenders summation value overlaps, preventing error-free computation through digital modulation [J2].

which is very difficult and impractical [30, 40].

ChannelComp can compute any function with a finite cardinality input domain, which is the case of any digital system due to the digital representation of values. As long as it is possible to count the output of the function, ChannelComp allows us to compute the desired function. For instance, to compute the function that returns the maximum of its input variables, the OAC method uses the log sum function to approximate the maximum, which returns an approximate value. By contrast, ChannelComp, for a finite input domain of the function, can calculate exactly the maximum of the given input variables.

Moreover, ChannelComp is not only compatible with digital modulation such as QPSK, QAM $4, 16, 64, \ldots$, but also FSK modulation, ASK, or PM, and other digital modulations. Furthermore, ChannelComp provides a wireless aggregation communication system as fast as OAC or even faster. This is because ChannelComp adapts the parameters of the digital modulation format such that the receiver computes the desired function, leading to a low latency computation over-the-air.

3.2 SumComp Coding: Digital Over-the-Air Computation via the Ring of Integers

This topic is investigated in Chapter 5, and is based on manuscript [J2], which introduces a novel and straightforward coding scheme, termed SumComp coding, using a ring of integers that enables digital modulations. The SumComp code is comprehensively compatible with various digital modulation forms, such as multiple levels of QAM and hexagonal QAM. Indeed, we present in this work a coding

method based on a ring of integers devised to circumvent the complexities associated with solving the optimization complexity of ChannelComp.

For instance, in Figure 3.2, we depict the output of SumComp code for QAM 16 versus Grady code for the same modulation. To diminish the complexity, we focus on computing only the summation function over-the-air, which allows us to compute a class of functions, known as nomographic functions [8], by adding pre- and post-coders to the system model. Through theoretical and numerical results, we demonstrate that our proposed coding scheme can be integrated into various digital modulation schemes. Additionally, we analyze the MSE associated with using SumComp coding in computing the arithmetic sum function. We also identify an upper limit for the mean absolute error (MAE) within a range of nomographic functions.

SumComp exhibits full compatibility, encompassing various forms of digital modulation. This includes QPSK, QAM at multiple levels (e.g., 4, 16, 64), hexagonal QAM, ASK, PM, and other digital modulation techniques. We establish the MSE for the SumComp code in case of computing the arithmetic sum function. Moreover, we provide an upper bound on the MAE for a class of nomographic functions. Different from the ChannelComp in [J1] and [C1], the SumComp code does not need to solve an optimization problem to obtain the modulation vector because it can provide a closed-form solution for the encoding procedure. Indeed, the SumComp code uses basic number theory concepts to map input value into a two-dimensional ring of integers, which allows us to perform the computation over-the-air.

In addition to the aforementioned advantages, our numerical experiments reveal that SumComp, mirroring the performance of ChannelComp, exceeds analog OAC in terms of computational accuracy across a range of crucial functions. In high signal-to-noise ratio (SNR) scenarios, SumComp demonstrates a roughly 10 dB enhancement over analog OAC, and orthogonal frequency division multiple access (OFDMA) methods for both sum and product functions.

3.3 Blind Federated Learning via Over-the-Air q-QAM

This topic is investigated in Chapter 6, and we present the content of manuscript [J3], which studies FEEL over a fading MAC wherein each ED participates in the training of a learning model by independently conducting computations using their own local data. For reducing the communication load, we use the proposed SumComp scheme in [J2] with QAM modulations, which leads to a low latency communication and spectral efficient scheme for the FEEL problem. Moreover, we employ multiple antennas at the edge server (ES) to compensate for the fading effects. We theoretically analyze the required number of antennas to diminish the effect of fading. Further, we derive MSE of the proposed scheme for both noisy and fading MAC. With the obtained MSE, we provide the convergence rate for a non-convex loss function in noisy and fading scenarios.

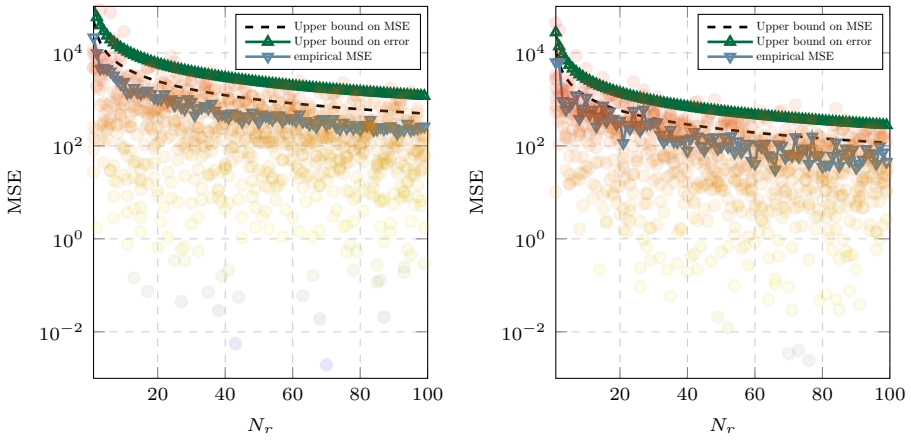
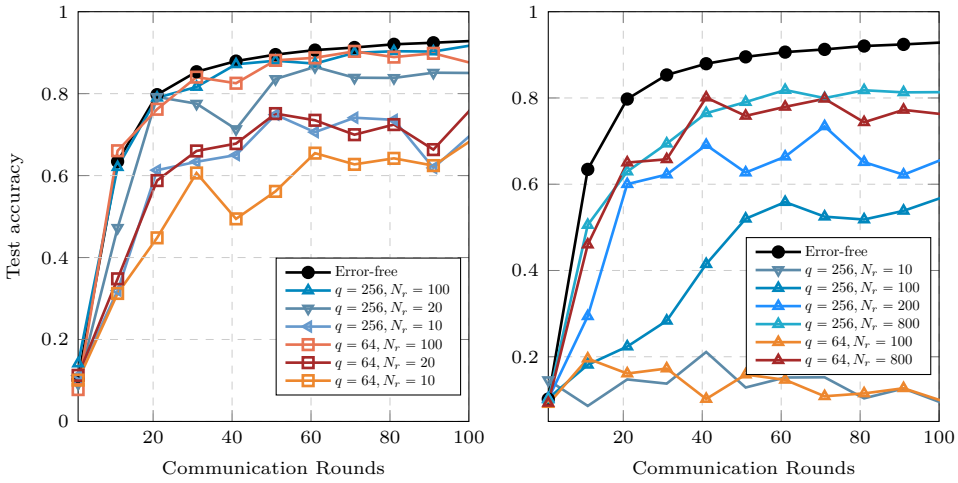


Figure 3.3: Monte Carlo numerical evaluation of the MSE of computing $\hat{s}^n$ from the measurement of the summation function for 10 trials versus the analytical results from Theorem 5 for $K = 200$ nodes and $\delta = 0.01$. The channel coefficients and channel noise generated by $N(0, \sigma_h)$ and $\mathcal{N}(0, \sigma_z)$, respectively, where $\sigma_h = \sigma_z = 1$. The figure shows empirical the MSE of $\hat{s}^n$, the analytical upper bound on the error in (6.25),and the expected value in (6.26) from Theorem 5 [J3].

Indeed, we employ the digital OAC scheme, SumComp from manuscript [J2], for FEEL problem, which is fully compatible with existing digital communication systems. The SumComp method improves the communication rate while providing reliable communication in FEEL network. In the proposed framework, we use multiple antennae at ES to compensate for the effect of fading. Therefore, EDs do not require knowledge of their channel state information (CSI). Further, ES only needs to estimate the sum of the channel coefficients of all the ED to each antenna, not each individual channel coefficients.

We provide the analysis for the required number of antennas N_r, at ES to obtain an upper bound for the error up to a certain level of the channel noise variance σ^2. In particular, we establish a probabilistic lower bound on the number of required antennas at ES and prove that the number of antennas has an inverse relation to the variance of the error, i.e., $N_r = \mathcal{O}(1/\sigma^2)$ (see Figure 3.3). We further derive the classic MSE analysis for both noisy and fading MAC conditions, considering QAM modulations of arbitrary order. Utilizing the derived MSE expressions, we provide a convergence analysis for non-convex loss functions, articulating this in terms of the MSE or the estimated gradient over the MAC.

In conclusion, our numerical analyses substantiate the theoretical assurances concerning MSE and the convergence efficacy of the digital FEEL framework. Notably, the results demonstrate that augmenting the number of antennas at the ES



(a) $(\sigma_z^2, \sigma_h^2) = (1, 1)$ (b) $(\sigma_z^2, \sigma_h^2) = (10, 1)$

Figure 3.4: Accuracy of the MNIST task as a function of the communication rounds for $K = 20$ ED devices and heterogeneous data distribution across ED. Figures 3.4a and 3.4b show the accuracy of FEEL versus number of communication rounds for two low variance of the noise, i.e., $\sigma_z^2 = 1$ and the high variance of the noise, i.e., $\sigma_z^2 = 10$, respectively [J3].

and adopting higher-order M-QAM facilitates superior estimation of average local model updates, enabled by improved signal alignment, reduced channel effects, and noise. In particular, Figure 3.4 shows that increasing the number of antennas at ES from $N_r = 10$ to $N_r = 100$, and the order of QAM modulations from $q = 64$ to $q = 256$, can lead to approximately 25% learning accuracy improvements in a fading channel.

3.4 Blind Asynchronous Over-the-Air Federated Edge Learning

We cover this topic in Chapter 7, which contains the content of manuscript [C2] by addressing the synchronization issue in OAC problem. Indeed, we targeted the problem of precoding the analog signals in FEEL, which is challenging to overcome any time misalignment at the receiver.

In this work, we propose a novel synchronization-free method to recover the learning parameters of the global model over-the-air without requiring any prior information about the time misalignments. We do not consider any synchronization overhead, such that the receiver has no prior information about the time misalignments of any participating user. We consider an analog OAC scheme where

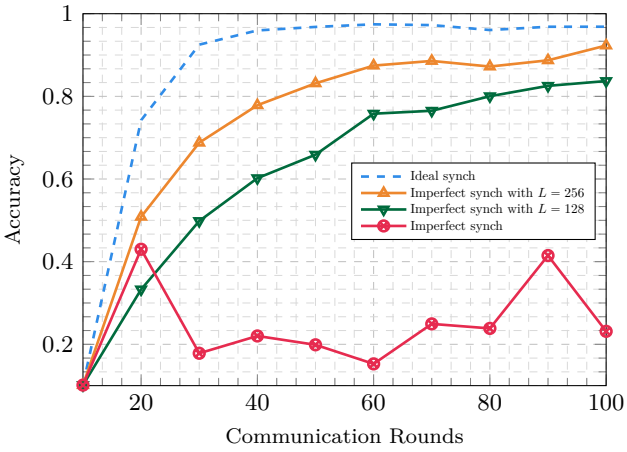


Figure 3.5: Accuracy of the MNIST task as a function of the communication rounds for $K = 10$ users and SNR 5 dB. The results show the performance of our proposed recovery method for imperfect and perfect synchronization [C2].

all devices transmit each parameter of their local model in an uncoded fashion and over a fading MAC, such that the receiver sees the superposition of asynchronous signals. We reformulate the measurement at the receiver in terms of atoms, and propose a convex optimization problem to recover the delays and the elements of the gradient, such that we obtain the summation to update the global gradient. We evaluate our proposed algorithm against the MNIST dataset and show how the proposed system achieves competitive results in this task, closing the performance gap down to 10% with respect to the ideal synchronized scenario and being 4× better in terms of learning accuracy than the simplest case where no recovery method is implemented (see Figure 3.5).

3.5 Contributions not Covered in the Thesis

In addition to the manuscripts listed above, I have worked on other topics and published other manuscripts during PhD studies. The following publications are not covered in the thesis but contain related materials and applications:

[C3] M. Bokaei, **S. Razavikia**, A. Amini and S. Rini, "Two-Snapshot DOA Estimation Via Hankel-Structured Matrix Completion," *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Singapore, Singapore, 2022, pp. 5018-5022.

[C4] M. Bokaei, **S. Razavikia**, A. Amini and S. Rini, "Single-snapshot DOA Estimation via Weighted Hankel-structured Matrix Completion," *IEEE Euro-*

pean Conference on Signal Processing(EUSIPCO), Belgrade, Serbia, 2022, pp. 1756-1760.

[J5] M. Seidi, **S. Razavikia**, S. Daei, and J. Oberhammer, "A Novel Demixing Algorithm for Joint Target Detection and Impulsive Noise Suppression," in *IEEE Communications Letters*, vol. 26, no. 11, pp. 2750-2754, Nov. 2022.

[C5] S. Daei, **S. Razavikia**, M. Kountoris, M. Skoglund, G. Fodor, C. Fischione, "Blind Asynchronized Goal-Oriented Detection for Massive Connectivity," *IEEE WiOpt Conference*, Singapore, Singapore, 2023.

[C6] **S. Razavikia**, S. Daei, C. Fischione, M. Skoglund, G. Fodor, "Off the Grid Blind deconvolution and remixing using atomic minimization," *IEEE Global Communications*, Kuala Lumpur, Malaysia, 2023.

[C7] H. Helstrom, **S. Razavikia**, V. Fodor, C. Fischione, "Optimal Receive Filter Design for Misaligned Over-the-Air Computation," *IEEE Workshop GLOBCOM*, Kuala Lumpur, Malaysia, 2023.

[C8] S. Hussein, **S. Razavikia**, S. Daei, C. Fischione, "Distributed Computing meets Matrix Factorization," submitted to *IEEE International Conference on Communications* 2024.

[J6] M. Bokaei, **S. Razavikia**, A. Amini and S. Rini, "Harmonic Retrieval Using Weighted Lifted-Structure Low-Rank Matrix Completion," in *Signal Processing Elsevier*, 2023.

[J7] **S. Razavikia**, M. Bokaei, A. Amini and S. Rini, "DOA Estimation via Weighted Hankel-structured Low-rank Matrix Completion," underreview *Signal Processing Elsevier*.

Contributions by the Author

The contributions of this Licentiate thesis' author on the mentioned manuscripts are the outcomes of the author's work in collaboration with the listed co-authors. The order of the author's name reflects the contribution level in the papers. The author of this Licentiate thesis, when being the first author of the paper, has been giving the substantial and vast majority of the contributions, especially in terms of ideas, theoretical analysis, computer simulations, and paper writing.

It is pertinent to acknowledge that the foundational concept of ChannelComp, which focuses on designing modulations by valid overlapping of constellation points, originated from the main supervisor. Additionally, the author of this Licentiate thesis has significantly contributed to the development and refinement of this idea throughout the research process.

3.6 Conclusions

Concurrently, the forthcoming 6G networks promise to open avenues for enhanced IoT applications supported by ML technologies. This burgeoning ecosystem, while indicative of technological progress, brings many challenges, especially the demand for communication rate protocols for existing communication systems due to escalated data traffic.

OAC has been identified as a viable solution to these impending challenges, with its innovative architecture enabling simultaneous data collection and computation at the network's edge. However, the reliance of OAC on analog communication accentuates inherent challenges, particularly concerning channel ramifications such as noise and fading effects, which could compromise signal integrity during transmission. The limited compatibility of analog modulations with contemporary wireless devices further exacerbates this challenge. Digital modulation, recognized for its robust channel correction and wide-ranging adoption across devices, presents itself as a compelling alternative, even though its integration with OAC remains a complex endeavor.

This research embarked on a groundbreaking journey by proposing and establishing the *ChannelComp* approach, a paradigm-shifting method that seamlessly integrates with prevailing digital communication systems. As we delved into its capabilities, it became evident that ChannelComp filled an important gap in the over-the-air approach; ChannelComp offers a universal computational framework more encompassing than the traditional OAC method limited to analog modulations. A standout feature of ChannelComp is its ability to efficiently manage a vast number of devices while rigorously adhering to computation time benchmarks.

Central to our thesis was the presentation of ChannelComp as an extension of OAC computation methodology capable of using digital and analog modulations to compute functions across MAC. Through strategic offline modulation selection techniques and tabular mappings, we redefined the potential of digital modulation in function computation over the MAC. Empirical evidence derived from simulation results attested to ChannelComp's superiority, showcasing its commendable performance enhancement versus the analog OAC and OFDMA methods, especially with a significant reduction in computation error.

Additionally, this research introduced the SumComp code, a simple yet potent coding mechanism based on a ring of integers. Besides addressing ChannelComp's computational complexities, the SumComp code also exhibited high adaptability; it was observed to be in synergy with a plethora of digital modulations, including multi-level QAM, Hexagonal QAM, and PM, among others. Moreover, we established the MSE for the SumComp code in case of computing the arithmetic sum function, an upper bound on the MAE for a class of nomographic functions.

Furthermore, we applied the ChannelComp protocol in FEEL to reach a communication efficient method using the SumComp code with M-QAM modulations, denoted as FedComp. As we expected, the resultant learning performance of FedComp shows robust performance in the presence of the fading and noise channel,

while the ED devices have no information of CSI. Besides, we analyzed the MSE and the convergence rate for FedComp and provided the theoretical guarantees regarding the number of antennas at ES for the MSE.

Besides the aforementioned investigation, we also addressed the time misalignment problem among nodes, which is one of the main challenges in the OAC technique for the FEEL application. We proposed a novel synchronization-free method to recover the parameters of the global model over the air without requiring any prior information about the time misalignment.

In summary, this thesis is a testament to the transformative potential of converging communication and computation. While ChannelComp represents a monumental stride, it does come with its intricacies. Yet, the presented methodologies and the SumComp code illuminate promising avenues bolstered by comprehensive MSE analyses and insights into MAE bounds. As we stand at this juncture, it is evident that our exploration has set the stage for future endeavors that can help redefine the horizons of digital wireless communications.

3.7 Future Works

Inspired by the potential of ChannelComp, we identify several promising directions for future research listed below:

- **Stochastic ChannelComp fading channel:** One avenue we intend to explore is the extension of ChannelComp to perform function computation based on CSI. This will involve adopting varying (both digital and analog) modulations for each node and a comprehensive evaluation of the implications of stochastic fading. Indeed, a rigorous analysis of the influence of channel statistical information on the optimization problem is needed.
- **Channel coding:** We have addressed a feasibility issue in manuscript [J1] by pinpointing modulation vectors conducive to efficient computation OAC. The proposed computational approach offers promising avenues for further optimization, such as minimizing maximal or average computational errors. There is potential to refine the proposed optimization paradigm concerning channel coding by incorporating additional time slots. This would allow for designing modulation vectors that exhibit enhanced robustness in the face of channel noise. Such a coding framework could be seamlessly integrated with the SumComp code, particularly within a deterministic setting, thus potentially contributing a novel channel code to the broader communication literature.
- **Time variant systems:** The paradigm of ChannelComp presents opportunities for adaptation, especially in designing or adjusting the modulation vector for a more generalized time-varying system model. Such adaptions could facilitate the computation of a series of functions over time or offer stabilization against channel fluctuations. While SumComp is adept at computing

summation functions, incorporating other ring algebraic structures, such as the polynomial ring and the quotient ring, might broaden the spectrum of functions that can be computed OAC.

- **MIMO extension for matrix computation:** Future endeavors also include enhancing the existing single narrowband antenna system, both at the transmitters and receiver ends, transitioning to broadband multiple-input and multiple-output systems. This transition would pave the way for vector-centric computations pertinent to various applications, encompassing matrix computation and federated learning.
- **Integrated communication and sensing:** Additionally, it would be instructive to explore the confluence of the ChannelComp methodology with the realm of integrated communication and sensing. Such an intersection offers opportunities to devise novel optimization strategies for fine-tuning modulation to serve dual objectives: computation and sensing.

References

[1] Mohammad Hassan Adeli and Alphan Şahin. Multi-cell non-coherent over-the-air computation for federated edge learning. In *IEEE Int. Conf. on Commun.*, pages 4944–4949, 2022.

[2] Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. SginSGD: Compressed optimisation for non-convex problems. In *International Conference on Machine Learning*, pages 560–569. PMLR, 2018.

[3] Léon Bottou and Yann Cun. Large scale online learning. *Advances in neural information processing systems*, 16, 2003.

[4] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, USA, 2004.

[5] R Creighton Buck. Approximate complexity and functional representation. Technical report, WISCONSIN UNIV MADISON MATHEMATICS RESEARCH CENTER, 1976.

[6] Bin Cao, Yixin Li, Lei Zhang, Long Zhang, Shahid Mumtaz, Zhenyu Zhou, and Mugen Peng. When internet of things meets blockchain: Challenges in distributed consensus. *IEEE Network*, 33(6):133–139, 2019.

[7] Mario Goldenbaum, Holger Boche, and Sławomir Stańczak. Harnessing interference for analog function computation in wireless sensor networks. *IEEE Trans. Sig. Proc.*, 61(20):4893–4906, 2013.

[8] Mario Goldenbaum, Holger Boche, and Sławomir Stańczak. Reliable computation of Nomographic functions over Gaussian multiple-access channels. In *ICASSP*, pages 4814–4818, 2013.

[9] Mario Goldenbaum, Holger Boche, and Sławomir Stańczak. Nomographic functions: Efficient computation in clustered Gaussian sensor networks. *IEEE Trans. Wireless Commun.*, 14(4):2093–2105, 2014.

[10] Mario Goldenbaum and Slawomir Stanczak. Robust analog function computation via wireless multiple-access channels. *IEEE Trans. on Commun.*, 61(9):3863–3877, 2013.

[11] Andrea Goldsmith. *Wireless communications*. Cambridge university press, 2005.

[12] Michael Grant and Stephen Boyd. CVX: MATLAB software for disciplined convex programming, version 2.1, 2014.

[13] Franz Hlawatsch and Gerald Matz. *Wireless communications over rapidly time-varying channels*. Academic press, 2011.

[14] Kaibin Huang. How does edge AI change the principles of communications and computing? Distinguished lecture, 2022, Nov, 14.

[15] Oltjon Kodheli, Eva Lagunas, Nicola Maturo, Shree Krishna Sharma, Bhavani Shankar, Jesus Fabian Mendoza Montoya, Juan Carlos Merlano Duncan, Danilo Spano, Symeon Chatzinotas, and Steven Kisseleff. Satellite communications in the new space era: A survey and future challenges. *IEEE Communications Surveys & Tutorials*, 23(1):70–109, 2020.

[16] Jiachun Li, Yan Meng, Lichuan Ma, Suguo Du, Haojin Zhu, Qingqi Pei, and Xuemin Shen. A federated learning based privacy-preserving smart healthcare system. *IEEE Transactions on Industrial Informatics*, 18(3), 2021.

[17] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of FedAvg on non-iid data. *arXiv preprint arXiv:1907.02189*, 2019.

[18] Gerald Matz and Franz Hlawatsch. Fundamentals of time-varying communication channels. In *Wireless communications over rapidly time-varying channels*, pages 1–63. Elsevier, 2011.

[19] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017.

[20] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

[21] Bobak Nazer and Michael Gastpar. Computation over multiple-access channels. *IEEE Trans. Info. Theo.*, 53(10):3498–3516, 2007.

[22] Quoc-Viet Pham, Rukhsana Ruby, Fang Fang, Dinh C Nguyen, Zhaohui Yang, Mai Le, Zhiguo Ding, and Won-Joo Hwang. Aerial computing: A new computing paradigm, applications, and challenges. *IEEE Internet Things J.*, 9(11):8339–8363, 2022.

[23] Saeed Razavikia, Sajad Daei, Mikael Skoglund, Gabor Fodor, and Carlo Fischione. Off-the-grid blind deconvolution and demixing. *arXiv preprint arXiv:2308.03518*, 2023.

[24] Saeed Razavikia, Jaume Anguera Peris, José Mairton B Da Silva, and Carlo Fischione. Blind Asynchronous Over-the-Air Federated Edge Learning. In *IEEE Globecom Workshops*, pages 1834–1839, 2022.

[25] Sebastian Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[26] David E Rumelhart, Geoffrey E Hinton, Ronald J Williams, et al. Learning internal representations by error propagation, 1985.

[27] Alphan Şahin, Bryson Everette, and Safi Shams Muhtasimul Hoque. Distributed learning over a wireless network with FSK-based majority vote. In *IEEE Int. Conf. on Advanced Commun. Technologies and Net.*, pages 1–9, 2021.

[28] Alphan Sahin, Bryson Everette, and Safi Shams Muhtasimul Hoque. Over-the-air computation with DFT-spread OFDM for federated edge learning. In *IEEE Conf. on Wireless Commun. and Net.*, pages 1886–1891, 2022.

[29] Alphan Sahin and Rui Yang. Over-the-air computation over balanced numerals. *arXiv preprint arXiv:2209.11004*, 2022.

[30] Alphan Sahin and Rui Yang. A survey on over-the-air computation. *arXiv preprint arXiv:2210.11350*, 2022.

[31] Alphan Sahin and Rui Yang. A survey on over-the-air computation. *IEEE Communications Surveys & Tutorials*, pages 1–1, 2023.

[32] Felix Sattler, Simon Wiedemann, Klaus-Robert Müller, and Wojciech Samek. Robust and communication-efficient federated learning from non-iid data. *IEEE transactions on neural networks and learning systems*, 31(9):3400–3413, 2019.

[33] Mohammadreza Seidi, Saeed Razavikia, Sajad Daei, and Joachim Oberhammer. A novel demixing algorithm for joint target detection and impulsive noise suppression. *IEEE Communications Letters*, 26(11):2750–2754, 2022.

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[35] David Sprecher. A representation theorem for continuous functions of several variables. *Proceedings of the American Mathematical Society*, 16(2):200–203, 1965.

[36] David A Sprecher. On the structure of continuous functions of several variables. *Trans. of the American Mathematical Society*, 115:340–355, 1965.

[37] Harsh Tataria, Mansoor Shafi, Andreas F Molisch, Mischa Dohler, Henrik Sjöland, and Fredrik Tufvesson. 6G wireless systems: Vision, requirements, challenges, insights, and opportunities. *Proc. IEEE*, 109(7):1166–1199, 2021.

[38] Ryan Tibshirani. *Notes on Stochastic Gradient Descent, Convex Optimization 10-725*. CMU university, USA, 2010.

[39] Joost Verbraeken, Matthijs Wolting, Jonathan Katzy, Jeroen Kloppenburg, Tim Verbelen, and Jan S Rellermeyer. A survey on distributed machine learning. *Acm computing surveys*, 53(2):1–33, 2020.

[40] Zhibin Wang, Yapeng Zhao, Yong Zhou, Yuanming Shi, Chunxiao Jiang, and Khaled B Letaief. Over-the-air computation: Foundations, technologies, and applications. *arXiv preprint arXiv:2210.10524*, 2022.

[41] Xiugang Wu, Shengli Zhang, and Ayfer Özgür. Stac: Simultaneous transmitting and air computing in wireless data center networks. *IEEE Journal on Selected Areas in Communications*, 34(12):4024–4034, 2016.

[42] Yong Zeng, Rui Zhang, and Teng Joon Lim. Wireless communications with unmanned aerial vehicles: Opportunities and challenges. *IEEE Communications magazine*, 54(5):36–42, 2016.

[43] Guangxu Zhu, Yuqing Du, Deniz Gündüz, and Kaibin Huang. One-bit over-the-air aggregation for communication-efficient federated edge learning: Design and convergence analysis. *IEEE Trans. Wireless Commun.*, 20(3):2120–2135, 2020.

[44] Guangxu Zhu, Yong Wang, and Kaibin Huang. Broadband analog aggregation for low-latency federated edge learning. *IEEE Trans. Wireless Commun.*, 19(1):491–506, 2019.

[45] Guangxu Zhu, Jie Xu, Kaibin Huang, and Shuguang Cui. Over-the-air computing for wireless data aggregation in massive IoT. *IEEE Wireless Commun.*, 28(4):57–65, 2021.

Part II

Included Papers