

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

Saeed Razavikia[✉], Member, IEEE, José Mairton Barros da Silva Jr.[✉], Member, IEEE,
and Carlo Fischione[✉], Senior Member, IEEE

Abstract—Communication and computation are traditionally treated as separate entities, allowing for individual optimizations. However, many applications focus on local information’s functionality rather than the information itself. For such cases, harnessing interference for computation in a multiple access channel through digital over-the-air computation can notably increase the computation, as established by the ChannelComp method. However, the coding scheme originally proposed in ChannelComp may suffer from high computational complexity because it is general and is not optimized for specific modulation categories. Therefore, this study considers a specific category of digital modulations for over-the-air computations, quadrature amplitude modulation (QAM) and pulse-amplitude modulation (PAM), for which we introduce a novel coding scheme called SumComp. Furthermore, we derive a mean squared error (MSE) analysis for SumComp coding in the computation of the arithmetic mean function and establish an upper bound on the mean absolute error (MAE) for a set of nomographic functions. Simulation results are presented to affirm the superior performance of SumComp coding compared to traditional analog over-the-air computation and the original coding in ChannelComp approaches in terms of both MSE and MAE over a noisy multiple access channel. Specifically, SumComp coding shows at least 10 dB improvements for computing arithmetic and geometric mean on the normalized MSE for low noise scenarios.

Index Terms—Constellation points, digital modulation, Gaussian integers, over-the-air computation, modulation coding, ring of integers.

I. INTRODUCTION

THE evolution of wireless communications, crucial to achieving the omnipresence of the Internet of Things

Manuscript received 26 October 2023; revised 14 April 2024 and 27 June 2024; accepted 19 August 2024. Date of publication 27 August 2024; date of current version 19 February 2025. Saeed Razavikia was supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP). José Mairton Barros da Silva Jr. was jointly supported by the European Union’s Horizon Europe research and innovation program under the Marie Skłodowska-Curie project FLASH, with grant agreement No 101067652; the Ericsson Research Foundation, and the Hans Werthén Foundation. The EU FLASH project, SSF SAICOM project, the Digital Futures project DEMOCRITUS, and the Swedish Research Council Project MALEN partially supported this work. The associate editor coordinating the review of this article and approving it for publication was Y. Fang. (*Corresponding author: Saeed Razavikia*.)

Saeed Razavikia is with the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, 114 28 Stockholm, Sweden (e-mail: sraz@kth.se).

José Mairton Barros da Silva Jr. is with the Department of Information Technology, Uppsala University, 751 05 Uppsala, Sweden (e-mail: mairton.barros@it.uu.se).

Carlo Fischione is with the School of Electrical Engineering and Computer Science and the Digital Futures, KTH Royal Institute of Technology, 114 28 Stockholm, Sweden (e-mail: carlofi@kth.se).

Digital Object Identifier 10.1109/TCOMM.2024.3450794

(IoT), has transitioned from human-centric to machine-type communications. The machine learning-based IoT applications in 6G [1], and the continuous growth of IoT devices [2], [3] facilitate handling vast data volumes. To alleviate such heavy communication burden, over-the-air computation (AirComp) has emerged as a solution for simultaneous data collection and computation at the network edge, exploiting the multiple access channel (MAC) waveform superposition property [4], [5].

AirComp is a non-orthogonal wireless communication method that optimizes communication rates and network coverage while preserving energy and data security. Notably, AirComp throughput scales linearly with the number of devices sharing the MAC [6], yielding significant gains for dense networks. Currently a hotbed of research in the wireless community, AirComp has potential applications in distributed machine learning, intra-chip communications, and wireless control [7]. However, AirComp is limited to analog amplitude modulation. Thus, we recently proposed a new communication for computation method termed ChannelComp [8], which allows the execution of arbitrary finite functions over the MAC and using digital modulations. ChannelComp enhancement broadens the scope of potential applications beyond the mere summation operations supported by the AirComp method.

However, due to its generality, ChannelComp’s coding scheme may suffer from computational complexity. Therefore, in this paper, we present a major advancement of the coding scheme for the ChannelComp method while preserving the advantages of low latency and spectral efficiency inherent to analog AirComp and ChannelComp. This enhancement in the coding scheme overcomes the limitations of analog AirComp’s feasibility and ChannelComp coding’s complexity and ensures full integration with existing digital communication systems.

A. Literature Review

The literature reveals two primary AirComp methodologies: uncoded analog aggregation [5], [9] and coded digital AirComp [10], [11]. The uncoded analog AirComp is extensively studied from various perspectives [4], [5], [12], [13], [14] and demonstrates significant resource efficiency, particularly in distributed learning, thus garnering attention for federated edge learning systems [15], [16]. The coded digital AirComp approach employs the nested lattice codes’ linearity for computation over Gaussian channels [17], [18].

However, AirComp’s dependency on analog communication presents reliability challenges due to channel ramifications,

e.g., noise and fading effects [19] and the requirements for analog hardware. With its superior channel correction properties and widespread adoption, digital modulation appears more favorable but poses difficulties due to the incoherence of overlapped digitally modulated signals [8], [20].

Recent developments in digital aggregation include methods such as one-bit broadband digital aggregation (OBDA) [21] and majority vote frequency-shift keying (FSK) [22]. Moreover, some variants have been proposed, such as the phase asynchronous orthogonal frequency-division multiplexing (OFDM)-based OBDA variant [23], and non-coherent communication solutions for AirComp [19], [22], [24]. For instance, in [24], the majority vote function was computed by nullifying a transmitter's contribution to the superposed value by encoding the votes into the zeros of a Huffman polynomial. These studies work based on the type-based multiple access principle, in which the frequency histogram is estimated for computing the average function by using orthogonal resources for different function output values [25], [26].

However, these studies mostly focused on specific functions (e.g., sign or summation function) or unique machine learning training processes (e.g., signSGD [27]). Proposals have used balanced number systems for summation functions, but with increased bandwidth usage [19]. Along this direction, an alternate encoding-based numeral system was introduced [28].

Furthermore, the authors in [29] utilize vector quantization to decrease the uplink communication overhead. They apply shared quantization and modulation codebooks specifically to address the federated edge learning problem. This method assumes that the number of transmitting devices is significantly less than the codebook size.

Despite these advancements, the usage of digital modulations with AirComp was limited to nomographic functions [18], [30] and simple digital modulations, such as BPSK or FSK. Indeed, these methods have enforced analog AirComp on digital communication, which was effective only in limited cases and often resulted in inefficient resource usage [31].

Recently, a new framework, denoted as ChannelComp [32], proposed general digital modulations for function computation. This framework aimed at addressing the issue of destructive overlaps of constellation points over the air. The groundbreaking idea of ChannelComp was to design digital modulation to circumvent destructive overlaps. In [33] and [34], channel coding techniques are applied based on the same principle to increase communication reliability.

Specifically, ChannelComp represents the constellation points of modulated signal x with a vector $\mathbf{x} \in \mathbb{C}^q$, where q represents the order of modulation. ChannelComp includes constraints to be met by vector \mathbf{x} to permit valid computation over the MAC. Given these constraints, ChannelComp poses the modulation design method as a feasibility problem to determine the digital modulation vector \mathbf{x} capable of ensuring a correct computation over-the-air. The resulting digital modulation allows the computation of the desired function using a tabular map. However, this optimization approach has high complexity when modulation order q is exceedingly high or when the number of nodes K is high [8].

In response to the complexity of the modulation design, we propose a new coding scheme with lower complexity

TABLE I
LIST OF COMMONLY USED VARIABLES HERE. ORDERED BY CASE AND ALPHABETICALLY

Variable	Definition
φ_k	Pre-processing function at node k (source encoder)
ψ	Post-processing function at the CP (source decoder)
\mathcal{E}_k	Join source-channel encoder at node k
\mathcal{D}	Join source-channel decoder
\mathcal{G}_ρ	Modulation mapper
\mathcal{E}_k	Encoder at node k
f	Desired function
\mathcal{D}	Decoder at the CP
K	Number of nodes in the network
q	Number of quantization level
r	Received signal by the CP
x_k	Modulated signal of node k
z	Additive white Gaussian noise
\mathcal{X}	The set of all constellation points

that maintains the full compatibility of ChannelComp with existing digital communication systems. We call such a coding scheme SumComp. It includes digital modulation techniques, such as quadrature-phase shift keying (QPSK), multi-level quadrature amplitude modulation (QAM) (e.g., 4, 16, 64), hexagonal QAM, and pulse-amplitude modulation (PAM). SumComp is an original coding scheme that can be included in the ChannelComp framework to reduce the modulation complexity design optimization problem, which was originally introduced in [32].

B. Contributions

We propose in this work a coding scheme, which we call SumComp, based on a ring of integers devised to circumvent the complexities associated with solving the optimization complexity of the modulation design for the ChannelComp over-the-air computation. To diminish the complexity, we focus on computing only the summation function over-the-air, which generalizes to a class of functions known as nomographic functions [30] by using pre and post-coders. The SumComp coding uses a ring of integers representing a 2D grid to compute the sum function over the MAC. Through theoretical and numerical results, we demonstrate that our proposed coding scheme can be integrated into various digital modulation schemes. Additionally, we analyze the mean squared error (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.

Note that while SumComp coding employs a structure that resembles a lattice in its use of a ring of integers representing a conceptual 2D grid, SumComp diverges fundamentally from traditional lattice coding. Traditional lattice codes map data within the same dimensionality (n-D to n-D), maintaining the dimension throughout the process. In contrast, SumComp coding operates as a bandwidth-expanding technique, mapping data from a lower dimension (1D) to a higher one (2D in our case). This expansion emphasizes the significance of the association of grid values rather than the transmitted values. The distinction and the methodological innovation of SumComp coding lie in its unique approach to dimensionality

and value association, setting it apart from conventional lattice coding strategies.

Specifically, our contributions are as follows:

- **Digital modulations:** We propose a novel digital modulation coding based on the ring of integers that can compute the sum function for a broad class of digital modulations. SumComp coding exhibits comprehensive compatibility, encompassing various forms of digital modulations. This includes QPSK, QAM at multiple levels (e.g., 4, 16, 64), hexagonal QAM, and other digital modulation techniques.
- **Joint source-channel coding:** We jointly design the encoding and constellation mapping schemes for the over-the-air computation problem. Specifically, the SumComp coding can be considered a joint source-channel coding for over-the-air computation, where a 1D input source signal is mapped into constellation points into a 2D (phase and quadrature components) signal towards computing a function. To the best of our knowledge, SumComp coding is the first joint source-channel coding scheme for digital over-the-air computation.
- **MSE analysis:** We establish the MSE for the SumComp coding in the case of computing the arithmetic sum function. Moreover, we provide an upper bound on the MAE for a class of nomographic functions.
- **Low complexity coding:** Different from the findings of [8], SumComp coding does not need to solve an optimization problem to obtain the modulation vector, due to its ability at providing a closed-form solution for the encoding procedure.
- **Numerical experiments:** In addition to the aforementioned advantages, our numerical experiments reveal that SumComp coding, mirroring the performance of ChannelComp, exceeds AirComp in terms of computational accuracy across a range of crucial functions. Notably, in high signal-to-noise ratio (SNR) scenarios, SumComp coding demonstrates a roughly 10 dB enhancement over other techniques for both sum and product functions.

C. Organization of the Paper

The rest of the paper is organized as follows: in Section II, we explain the system model, including the problem statement and the signal model. Next, we describe in detail the architecture of the encoder and decoder of the proposed SumComp coding in Section III. In Section IV, we analyze the performance of the SumComp encoder for the sum function and the class of nomographic functions in terms of MSE error and MAE error, respectively, and provide theoretical upper bounds for the MSE and MAE metrics. We present the numerical results for the theoretical bounds and the performance analysis between SumComp coding, the original coding proposed in ChannelComp, AirComp, and the traditional orthogonal frequency division multiple access (OFDMA) communication in Section V. Finally, we conclude the paper in Section VI.

D. Notation

We denote a finite field by \mathbb{F}_q , where q denotes the number of elements inside the field. Moreover, we denote by \mathbb{Z} , \mathbb{R} , and

\mathbb{C} as the integer, real, and complex number sets, respectively. Let ρ be a non-zero complex number. We define the ring $\mathbb{Z}[\rho]$ as $\mathbb{Z}[\rho] = \{a + b\rho \mid a, b \in \mathbb{Z}\}$. Then, we denote by \mathbb{G} the Gaussian integer set that is $\mathbb{Z}[i]$, i.e., a Gaussian integer $z = a + bi \in \mathbb{G}$, is a complex number whose real and imaginary parts are integer values, $\{a + bi \mid a, b \in \mathbb{Z}\}$. Moreover, let q_1 and q_2 be integers relatively prime integers or *co-prime*, i.e., their greatest common divisor is $\gcd(q_1, q_2) = 1$. Then, by Bézout's identity, there exist integers μ_1 and μ_2 such that $q_1\mu_1 + q_2\mu_2 = 1$.

We use lowercase letters x for scalar and calligraphic notation \mathcal{X} to represent operators. For a complex scalar $x \in \mathbb{C}$, $\Re(x)$ and $\Im(x)$ denote the real and imaginary parts of x , respectively. Moreover, we use $\mathbb{E} \subset \mathbb{R}$ as the closed unit interval. $\mathcal{F}(\mathbb{A})$ denotes the space of every function $f : \mathbb{A} \mapsto \mathbb{R}$ for some topological space \mathbb{A} . Moreover, $\mathcal{C}^0(\mathbb{A})$ denotes the space of real-valued continuous functions with domain \mathbb{A} . Furthermore, $\mathcal{CN}(0, \sigma^2)$ denotes a circularly symmetric complex normal distribution, wherein the real and imaginary terms are each distributed according to a normal distribution with variance σ^2 , i.e., $\mathcal{N}(0, \sigma^2)$.

II. SYSTEM MODEL

The system model is identical to the one on ChannelComp [8], except that we introduce an improved coding scheme and restrict ourselves to nomographic functions. We study a network composed of K nodes and a computational server, termed the computation point (CP), connected through a common communication channel. This architecture is leveraged to compute a predetermined function $f(s_1, s_2, \dots, s_K)$ at the CP. Each node k owns a unique input $s_k \in \mathbb{F}_q$, which needs to transmit their s_k values via digital communication. In this system, nodes simultaneously transmit over the MAC to enable the computation of f at the CP.

In this digital transmission scenario, each input $s_k \in \mathbb{F}_q$ undergoes an encoding block through $\mathcal{E}_k(\cdot)$ to produce a digitally modulated signal x_k , with $x_k = \mathcal{E}_k(s_k)$, where $x_k \in \mathbb{C}$ is a complex value whose real and imaginary parts correspond to in-phase and quadrature components. Each node k then transmits x_k over the communication channel (see Figure 1).

Due to the synchronous transmission of all nodes over identical frequencies or codes,¹ the CP server obtains the sum of all x_k 's through the MAC within a single time slot, as represented in equation 1, i.e.,

$$r = \sum_{k=1}^K h_k p_k x_k + z \in \mathbb{C}. \quad (1)$$

Here, r represents the superimposed electromagnetic waves, h_k refers to the channel coefficient between node k and the CP, p_k indicates node k 's transmission power, and z is the inherent receiver noise. The receiver noise, z , is typically modeled as an additive white Gaussian noise process with zero-mean and variance σ^2 , which is circularly symmetric, i.e., $z \sim \mathcal{CN}(0, \sigma^2)$. Following the power control universally adopted

¹It is assumed that the synchronization among all the nodes and the CP is perfect. Imperfect synchronization can be handled by using the existing techniques, e.g., [35] and [36].

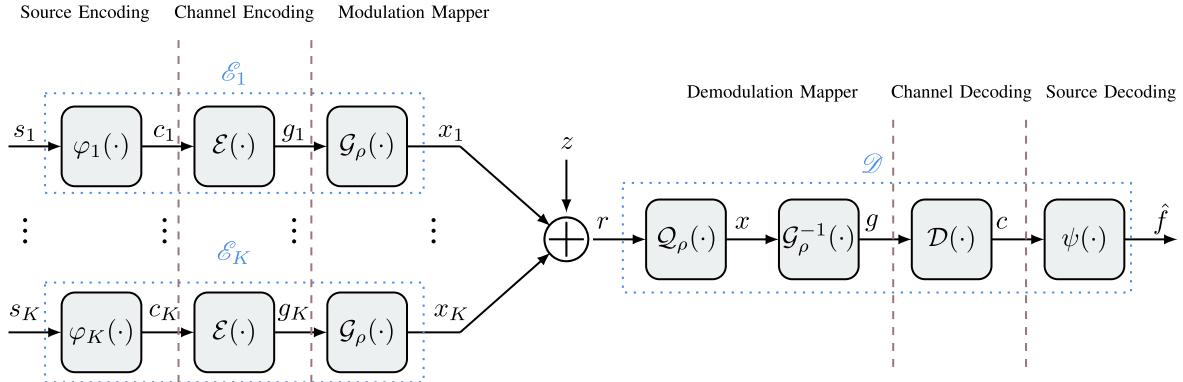


Fig. 1. Block diagram illustrating the complete transmission process for computing a function over the MAC with K nodes. The process begins with the signals s_1, s_2, \dots, s_K being passed through an encoder $\mathcal{E}(\cdot)$ and decoder \mathcal{D} . The modulated signals x_1, x_2, \dots, x_K are then transmitted over the MAC, leading to r contaminated by the noise z . The received signal r undergoes quantization $\mathcal{Q}(\cdot)$. Then, it is passed through an inverse function \mathcal{G}_ρ^{-1} to obtain $g \in \mathbb{G}$, which is finally decoded by the decoder $\mathcal{D}(\cdot)$ and processed by function $\psi(\cdot)$ to yield the estimated function \hat{f} . The dashed boxes denote the Encoding region \mathcal{E} and the Decoding region \mathcal{D} . In a standard communication system, φ , \mathcal{E} , and \mathcal{G}_ρ correspond to source coding, channel encoding, and modulation mapper blocks, respectively. Similarly, ψ , \mathcal{D} , \mathcal{G}_ρ^{-1} corresponds to source decoding, channel decoding, and demodulation mapper blocks, respectively.

in the over-the-air literature [37], we select the transmit power as the inverse of the channel, i.e., $p_k = h_k^*/|h_k|^2$ for $k \in [K]$. Hence, Eq. (1) simplifies to

$$r = \sum_{k=1}^K x_k + z. \quad (2)$$

In the following, we use Eq. (2) without losing generality.

Finally, the estimation of the desired function f can be obtained via a proper decoding scheme \mathcal{D} at the CP, i.e.,

$$\hat{f} := \mathcal{D}(r) = f. \quad (3)$$

Notably, this system model parallels the AirComp and ChannelComp [8] system models over the MAC, with the only difference being the digital modulation. Thus, this model supports low-latency communication across numerous nodes, as evidenced by [20]. Furthermore, the encoding and constellation mapping schemes are jointly designed for the over-the-air computation problem.

The goal of this paper is to devise the encoders $\mathcal{E}_k(\cdot)$ and the decoder $\mathcal{D}(\cdot)$ to efficiently compute the desired function f at the CP by harnessing the superposition in (2). The main challenges are as follows:

- In the absence of noise, the received signal r by the CP has a reshaped finite constellation diagram, i.e., it is a deformation of the original constellation diagram of the transmitting nodes x_k for $k \in [K]$. Then, decoding the desired function f from overlapped constellation signals becomes challenging [32].
- The superposition of channel input symbols is corrupted by the Gaussian noise.
- Identifying a set of the encoders $\mathcal{E}_k(\cdot)$ and the decoder $\mathcal{D}(\cdot)$ for general function computation, like in ChannelComp, is non-trivial. Thus, we limit the function to the specific class of nomographic functions for closed-form computation.
- Direct design of the encoders $\mathcal{E}_k(\cdot)$ and corresponding decoder is intricate. A modular approach is adopted for separate design (refer Figure 1).

In the next section, we propose an architecture for the encoder $\mathcal{E}(\cdot)$ and decoder $\mathcal{D}(\cdot)$ in detail.

III. SUMCOMP: ENCODING AND DECODING METHODS

This section describes how to design the encoder and decoder of our proposed coding scheme, termed SumComp. Before proceeding, we need to briefly recap some necessary definitions and mathematical background provided in the sequel that help the reader understand the results of this paper.

A. Definitions and Motivation

Here, we target a class of functions called a *nomographic* functions or a set of nomographic functions defined as follows.

Definition 1: [Nomographic function [7]]. Let \mathbb{S}^K , for $K \geq 2$, be a compact metric space. A function $f : \mathbb{S}^K \mapsto \mathbb{R}$ for which there exist pre-processing functions $\varphi_k : \mathbb{S} \mapsto \mathbb{R}$ for $k \in [K]$ and post-processing function $\psi : \mathbb{R} \mapsto \mathbb{R}$ such that f can be represented by

$$f(s_1, \dots, s_K) = \psi \left(\sum_{k=1}^K \varphi_k(s_k) \right), \quad (4)$$

is called a nomographic function.

Note that the compactness mentioned in Definition 1 is a critical assumption regarding the commuting of the continuous functions. Indeed, let \mathcal{N} , $\mathcal{N}^0(\mathbb{E}^K)$ be the space of nomographic functions and the space of nomographic functions with the restriction of continuous pre- and post-processing functions, respectively. Then the following interesting result is due to Sprecher and Buck.

Theorem 1: [Sprecher'65 [38]]. Every function $f \in C^0(\mathbb{E}^K)$ can be represented with real, monotonic increasing pre-processing functions and possibly a discontinuous post-processing function.

Theorem 2: [Buck'79 [39]]. Every function $f \in \mathcal{F}(\mathbb{E}^K)$ is nomographic, (i.e., $\mathcal{N}(\mathbb{E}^K) = \mathcal{F}(\mathbb{E}^K)$).

If one desires the pre- and post-processing functions to be continuous for an arbitrary continuous function f , Theorem 2 is unfortunately not valid. In this case, one may construct

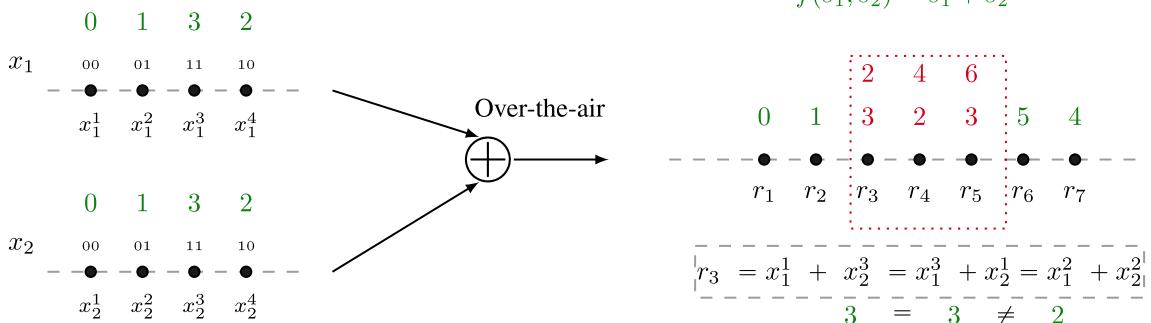


Fig. 2. Constellation diagram showcasing the transmission of Gray-coded PAM-modulated signals by two nodes ($K = 2$) with $q = 4$ (2 bits). Destructive overlaps, depicted in red, occur when transmitting values $\{3, 2, 3\}$ and $\{2, 4, 6\}$ concurrently, demonstrating the challenges in uniquely decoding the received signals due to interference in the same channel.

a nomographic function approximating the desired function. For example, under Definition 1, the geometric mean on \mathbb{E}^K is a function that can be approximated with $\varphi_k(s) = \ln(s + 1/p_0(\epsilon))$ and $\psi(s) = \exp(s/K)$ for $p_0(\epsilon) > 0$. For a comprehensive discussion of nomographic functions and their properties, we refer the reader to [7], [30], and [40].

To compute the nomographic functions, each value $s_k \in \mathbb{E}$ is first encoded to the corresponding pre-processing function φ_k to produce the output value $c_k := \varphi_k(s_k) \in \mathbb{R}$. Because the Gaussian MAC (2) is a finite capacity channel, communicating the real c_k over such a channel with infinite precision is impossible. Hence, we have to first quantize c_k to finite precision value with q different levels, i.e., $c_k \in \mathbb{F}_q$ for $k \in [K]$. Without loss of generality, we assume the range of c_k 's are uniformly quantized so that \mathbb{F}_q can be mapped to \mathbb{Z}_q uniquely using a bijective map.² Next, adding the post-processing function $\psi(\cdot)$ at the end of the decoding scheme, we are able to compute the desired nomographic function f (see Figure 1).

In the subsequent phase, there is a requirement to map the $c_k \in \mathbb{Z}_q$ onto constellation points $x_k \in \mathbb{Z}^2$ in preparation for transmission. Under typical communication scenarios, Gray coding can map the q potential values of c_k to the constellation points x_k , e.g., QAM modulation of order q . Subsequently, x_k is transmitted via the MAC to compute the function f . However, as demonstrated in [8], the utilization of Gray code in high-order modulation induces destructive interference in the resulting constellation points $\sum_{k=1}^K x_k$. This ultimately leads to an inability to compute the function f from r using a decoding scheme \mathcal{D} , implying $\hat{f} \neq f$ for any given \mathcal{D} .

To provide a more illustrative example, we present a simple computation scenario for the sum function $f(s_1, s_2) = s_1 + s_2$ for $K = 2$ nodes with a Gray-coded PAM 4 in Figure 2. The constellation points of Gray-coded PAM can be obtained using the following relation:

$$x_k = (s_k + \lfloor s_k/2 \rfloor - 2\lfloor s_k/3 \rfloor - 3)E_s/2, \quad k = 1, 2, \quad (5)$$

where E_s denotes the amplitude of the carrier signal. Then, the constellation points for the $x_1 + x_2$ are given by $r_i = (i - 4) \times E_s$ for $i \in \{1, \dots, 7\}$. Note that the constellation points r_3, r_4 , and r_5 need to be assigned simultaneously to

²Since the domain of s_k is compact, consequently, the range of preprocessing function φ_k is a compact interval. Moreover, the union of these intervals becomes compact as well. As a result, every value in the range of c_k 's has a unique dyadic expansion and can be approximated by terminating the dyadic expansion [18].

different values. Specifically, r_3 corresponds to both 3 and 2, r_4 corresponds to both 2 and 4, and r_5 corresponds to both 3 and 6. Therefore, the sum function cannot be computed because the conflict occurs on the abovementioned points. Specifically, in scenarios where nodes 1 and 2 transmit respective values of $s_1 = 1$ and $s_2 = 1$ through the MAC, the CP discerns the constellation point r_3 . Similarly, the transmission of $s_1 = 0$ and $s_2 = 3$ by the nodes also results in the CP perceiving the same constellation point r_3 . This poses a challenge in the computation of the aggregate $s_1 + s_2$ at the CP, as it necessitates assigning the constellation point r_3 to two distinct sums, 3 and 2, which is not feasible. Such overlaps prevent precise computation of the desired sum at the CP, showing the inherent limitation of traditional digital AirComp.

In the following Section III-B, we introduce the working principle of the SumComp encoder and how it can solve destructive overlaps.

B. SumComp Encoder

To encode the constellation points of the modulation x_k to avoid overlaps, we first introduce an encoding scheme that maps any points from \mathbb{Z} to a set of points (line) in a general integer ring $\mathbb{Z}[\rho]$. Then, we show that the induced sets of points belong to a group set under the addition operator. Therefore, the resultant superposition of the constellation points results in a new set of points that belongs to the same set and can be uniquely computed by the summation. Note that the integer set $\mathbb{Z}[\rho]$ helps us to express a general two-dimensional modulation with in-phase and quadrature components.

As outlined in the system model section, the computation of general functions through over-the-air processing poses challenges. A modular approach is employed to circumvent these challenges, specifically targeting nomographic functions. This approach simplifies the computational process by allowing for the independent computation of pre-processing and post-processing functions, thus reducing the requirement to compute the summation function over the air.

Several key factors must be considered regarding the design of an encoder for the summation function. The symmetric nature of the summation function permits the utilization of the same encoder across all nodes, represented as $\mathcal{E}_k(\cdot) = \mathcal{E}(\cdot)$. While it is not necessary, the encoder's ability to operate as an additive map offers considerable benefits, ensuring that $\sum_{k=1}^K \mathcal{E}(s_k) = \mathcal{E}\left(\sum_{k=1}^K s_k\right)$. Consequently, the objective is

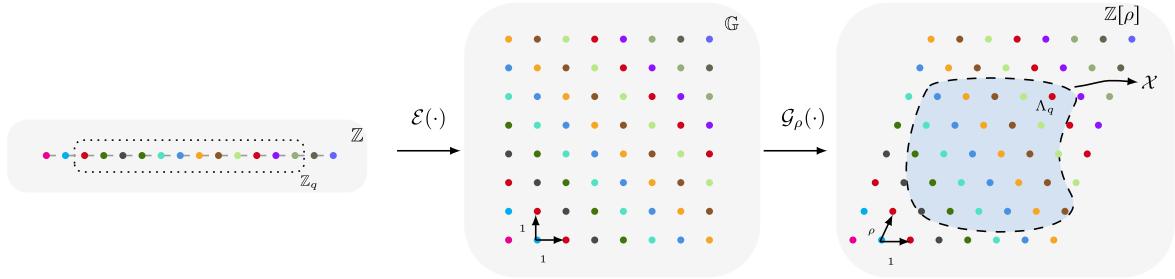


Fig. 3. The complete encoding procedure. The output of preprocessing function $c_k \in \mathbb{Z}_q$ is first encoded to Gaussian integers \mathbb{G} using the encoder $E(\cdot)$. Then, using $G_\rho(\cdot)$, we map the output value to a general ring of integers $\mathbb{Z}[\rho]$. Finally, node k selects subset Λ_q of $\mathbb{Z}[\rho]$ as the final constellation points to transmit over the MAC.

to identify a mapping $\mathcal{E} : \mathbb{Z} \mapsto \mathbb{G}$ that enables the output $g_k := \mathcal{E}(s_k)$, in conjunction with the input value s_k , to form an additive group.

We define encoder operator $\mathcal{E}_{q_1, q_2} : \mathbb{Z} \mapsto \mathbb{G}$ with fixed co-prime integers (q_1, q_2) for given input integer value $m \in \mathbb{Z}$ as

$$g_k := \mathcal{E}_{q_1, q_2}(m) = m\mu_1 + kq_2 + (m\mu_2 - kq_1)i \in \mathbb{G}, \quad (6)$$

for $k \in \mathbb{Z}$, where μ_1 and μ_2 are integers such that $1 = q_1 \cdot \mu_1 + q_2 \cdot \mu_2$; they are obtained using the *extended Euclidean algorithm* [41]. The encoder function $\mathcal{E}_{q_1, q_2}(\cdot)$ is defined in terms of q_1 and q_2 ; integers μ_1 and μ_2 that satisfy Bézout's identity for q_1 and q_2 ; and k which varies over all integers. The encoder is an isomorphism map that makes the integers number in \mathbb{Z} and lines in Gaussian integer ring \mathbb{G} isomorphic.

Remark 1: The encoder \mathcal{E}_{q_1, q_2} serves the purpose of channel coding within communication systems, introducing redundancy by mapping input value points to multiple points (or a line). This added redundancy contributes to the system's resilience against channel noise and facilitates error control, effectively expanding the bandwidth, as elucidated by [42].

Remark 2: Due to additive map \mathcal{E}_{q_1, q_2} , it is possible to apply the standard channel codes, e.g., non-binary LDPC, and error correction schemes to the SumComp codes to combat the synchronization errors [34]. Furthermore, for the sample-level synchronization, it is possible to design the matched filter to mitigate the error, as studied in [43] and [36].

Next, for the sake of generality, we map the encoded value g_k into a general ring of integers $\mathbb{Z}[\rho]$ using an operator $G_\rho : \mathbb{G} \mapsto \mathbb{Z}[\rho]$ for $\rho \in \mathbb{C}$. Indeed, for a given $g = a + bi$, we have

$$x = G_\rho(a + bi) = a + b\rho i, \quad a, b \in \mathbb{Z}. \quad (7)$$

Then, it is clear that the linear map G_ρ is an isomorphism from \mathbb{G} to the ring of integers $\mathbb{Z}[\rho]$. Now, we can define the constellation points as a finite subset of generated points in $\mathbb{Z}[\rho]$. Specifically, let \mathcal{X} denote the constellation points generated through the modulation scheme applied to our source messages c_k , for $k \in [K]$. To have a finite set of constellation points, we need to restrict the generated lines by encoder \mathcal{E}_k to a given grid subset $\Lambda_q \subset \mathbb{Z}[\rho]$ with cardinality $|\Lambda_q| \geq q$, for $k \in [K]$. Mathematically, this relationship is formulated as follows:

$$\mathcal{X} = \{G_\rho(\mathcal{E}_{q_1, q_2}(c)) \mid c \in \mathbb{Z}_q\} \cap \Lambda_q. \quad (8)$$

This implies that the constellation points can be any arbitrary subset of infinitely generated points by \mathcal{E}_k from $c \in \mathbb{Z}_q$ (see

Figure 3). Consider a point in the domain \mathbb{Z}_q , represented as a discrete line in the complex domain \mathbb{G} . These corresponding points in \mathbb{Z}_q and \mathbb{G} share the same color in Figure 3. The mapping G_ρ then translates these discrete lines into the ring of integers, denoted by $\mathbb{Z}[\rho]$. Finally, a finite subset of these points can be selected to constitute a constellation diagram \mathcal{X} (black points in Figure 3).

Remark 3: Equation (8) is sufficiently general to encompass a broad class of digital modulation by selecting an apt grid subset Λ_q . Two degrees of freedom are evident. Firstly, G_ρ permits flexible grid generation by adjusting $\rho \in \mathbb{C}$. Secondly, the subset Λ_q allows for diverse modulation selections through various grid subsets.

Remark 4: It is important to clarify that while \mathcal{X} employs a ring of integers $\mathbb{Z}[\rho]$, SumComp coding significantly differs from traditional lattice coding approaches. Unlike traditional lattice codes, where grid values carry inherent meaning, SumComp emphasizes the association of these values as determined by the encoder \mathcal{E}_{q_1, q_2} . Thus, the relationship among grid values, rather than the values themselves, are of primary importance in our scheme. This fundamental shift from value-centric to association-centric encoding [32] distinguishes SumComp from conventional lattice-based methods.

In what follows, we discuss some examples of the encoder scheme.

Example 1: For QAM with order q , which is a perfect square number, the subset Λ_q is simply the Cartesian product of $\{1, \dots, \sqrt{q}\}$ with itself. Next, we set $q_2 = \sqrt{q}$ and $q_1 = 1$ and $\mathcal{G}_i = \mathcal{I}$, where \mathcal{I} is the identity operator. In particular, the SumComp encoders $\mathcal{E}_k(\cdot)$ become $\mathcal{E}_{\sqrt{q}} : \mathbb{Z}_q \mapsto \mathbb{G}$. For input $c \in \mathbb{Z}_q$, we have

$$\begin{aligned} \mathcal{X} &= \{\mathcal{E}_{1, \sqrt{q}}(c) \mid c \in \mathbb{Z}_q\}, \\ \mathcal{E}_{\sqrt{q}}(c) &:= c - \sqrt{q} \cdot \left\lfloor \frac{c}{\sqrt{q}} \right\rfloor - \sqrt{q} + \left(\left\lfloor \frac{c}{\sqrt{q}} \right\rfloor - \sqrt{q} \right) i \in \mathbb{G}. \end{aligned} \quad (9)$$

The map is bijective. For instance, the encoded values of QAM with order $q = 16$ are depicted in Figure 5.

Example 2: For PAM modulation with order q , which is an odd number, the subset Λ_q is defined as $\Lambda_q = \{-\lfloor q/2 \rfloor, \dots, 0, \dots, \lfloor q/2 \rfloor\}$ with $q_1 = 1$, $q_2 = q$, and $\mathcal{G}_i = \mathcal{I}$. The SumComp encoder $\mathcal{E}_q(\cdot) : \mathbb{Z}_q \mapsto \mathbb{Z}_q$ becomes

$$\mathcal{X} = \{\mathcal{E}_{1, q}(c) \mid c \in \mathbb{Z}_q\}, \quad \mathcal{E}_q(c) := c - \lfloor q/2 \rfloor, \quad (10)$$

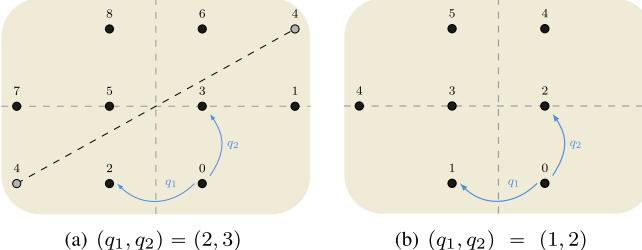


Fig. 4. SumComp coded Hexagonal QAM of order 8 for two different choices of (q_1, q_2) . In Figure 4(a), $(q_1, q_2) = (2, 3)$ shows that the input value s_k can have integer values between 0 and 8, except for 4. Adding the value 4 requires new constellation points on one of the gray constellation points in the dashed lines, avoiding overlaps. In Figure 4(b), the SumComp coded Hexagonal QAM 8 with $(q_1, q_2) = (1, 2)$ shows that the input value s_k can have integer values between 0 and 5. Here, the numbers 4 and 1 are repeated. Replacing the numbers 4 or 1 with any other numbers results in overlaps among the constellation points. Note that for the repeated constellation points, every node needs to select one of the constellation points for transmission.

and the decoder $\mathcal{D}_q(x) : \mathcal{X} \mapsto \mathbb{Z}_q$ is defined as

$$\tilde{m} := \mathcal{D}_q(x) := \Re(x) + K\lfloor q/2 \rfloor, \quad \forall x \in \mathcal{X}. \quad (11)$$

It is worth mentioning that the subset Λ_q does not need to be rectangular, and any subset of the generated two-dimensional grid by the encoder $\mathcal{E}(\cdot)$ can be selected. This can be illustrated by Figure 4, where we depict the SumComp coded constellation diagram with two different choices of (q_1, q_2) for Hexagonal QAM modulation. In Figure 4(a), the Hexagonal QAM 8 uses $\mathcal{E}_{2,3}$ for $s_k \in \{0, \dots, 3, 5, \dots, 8\}$. Adding the value 4 requires new constellation points in the dashed lines on one of the Gray constellation points to avoid overlaps. Therefore, we need to either increase the modulation order or change the choice of (q_1, q_2) in Figure 4(b). For the case of $(q_1, q_2) = (1, 2)$, the input domain reduced to the input value $s_k \in \{0, \dots, 5\}$. These examples demonstrate that for a given constellation diagram (subset Λ_q), the encoder \mathcal{E} can be a non-surjective mapping. For instance, in Figure 4(b), the value 1 is assigned to two different constellation points. Each node must select only one of the repeated constellation points for transmission to ensure a bijective mapping.

The constellation diagrams presented in Examples 1 and 2 are not optimized for energy efficiency due to their asymmetry about the origin. However, adjusting the entire constellation diagram with a complex constant can provide symmetry and improve energy efficiency.

Remark 5: Note that the encoder's output, $\mathcal{E}(\cdot)$, can undergo a shift by any complex scalar, $\gamma_1 \in \mathbb{C}$, and multiplication with $\gamma_2 \in \mathbb{C}$. Consequently, the decoder, \mathcal{D}_q , must adjust the received value by multiplying $\gamma_2^*/|\gamma_2|$ and then subtracting $K\gamma_1$ to maintain the integrity of the computation process, thereby ensuring the successful computation of the summation.

Note that for computing function f over the MAC, we must show that the constellation points are closed under addition. To this end, the assigned values to constellation point $x \in \mathcal{X}$ by encoder $\mathcal{E}_k(\cdot)$ form an additive group, which we call the SumComp modulation group. Consequently, additive MAC can not make destructive overlaps as it happens with the Gray code. In particular, we define the SumComp modulation group in Definition 2.

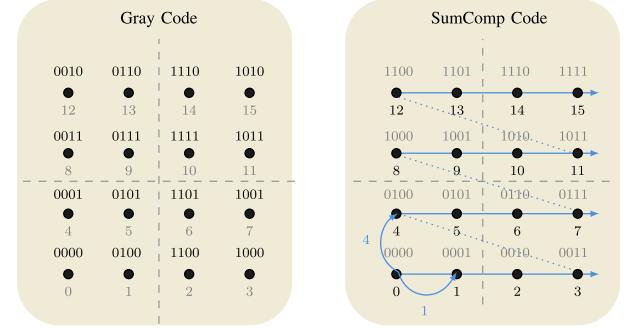


Fig. 5. Gray code vs SumComp code for QAM 16 modulation with $(q_1, q_2) = (1, 4)$.

Definition 2: For a given ring of integers $\mathbb{Z}[\rho]$, the SumComp modulation set $\mathcal{S}_{q_1, q_2}^\rho$ is defined with a precode operator \mathcal{G}_ρ and fixed co-prime integers $q_1, q_2 \in \mathbb{Z}$ as follows

$$\begin{aligned} \mathcal{S}_{q_1, q_2}^\rho := & \left\{ (c, x) \mid x \in \mathbb{Z}[\rho], c \in \mathbb{Z}, \right. \\ & \left. c = \Re(\mathcal{G}_\rho^{-1}(x))q_1 + \Im(\mathcal{G}_\rho^{-1}(x))q_2 \right\}, \end{aligned} \quad (12)$$

where \mathcal{G}_ρ^{-1} is an isomorphism that maps points of the ring of integers $\mathbb{Z}[\rho]$ back to the Gaussian integer set \mathbb{G} .

In the following, we prove that $\mathcal{S}_{q_1, q_2}^\rho$ is a group under the addition.

Proposition 1: The SumComp modulation set $\mathcal{S}_{q_1, q_2}^\rho$ is an additive group.

Proof: See Appendix A. \square

Note that $\mathcal{S}_{q_1, q_2}^\rho$ is a group whose elements are the values of input signal c with its corresponding constellation points x . Proposition 1 proves that $\mathcal{S}_{q_1, q_2}^\rho$ is a group under addition, which means that the summation $\sum_{k=1}^K c_k$ with corresponding pair $\sum_{k=1}^K x_k$ also belongs to $\mathcal{S}_{q_1, q_2}^\rho$. As a result, using a proper isomorphic decoding scheme \mathcal{D} , we assign $\sum_k x_k$ to $f = \sum_k s_k$ and compute the desired summation function.

The next section explains the overall decoding procedure to compute the function f .

C. SumComp Decoder

After transmitting x_k 's by nodes, the point r in $\mathbb{Z}[\rho]$ deviates due to channel noise and no longer belongs to the same integer grid. To map $r \in \mathbb{C}$ back to the grid $\mathbb{Z}[\rho]$, we use the associated quantizer operator to the ring of integers $\mathbb{Z}[\rho]$ denoted by $\mathcal{Q}_\rho : \mathbb{C} \mapsto \mathbb{Z}[\rho]$ and defined as

$$\mathcal{Q}_\rho(\mu) = \operatorname{argmin}_{x \in \mathbb{Z}[\rho]} \|\mu - x\|_2, \quad (13)$$

where \mathcal{Q}_ρ assigns every $\mu \in \mathbb{C}$ to the nearest point, with respect to the Euclidean distance, the ring of integers $\mathbb{Z}[\rho]$. Hence, using \mathcal{Q}_ρ , we note that $x = \mathcal{Q}_\rho(r) \in \mathbb{Z}[\rho]$ lies on the ring $\mathbb{Z}[\rho]$. Then, the recovered point x from $\mathbb{Z}[\rho]$ is mapped to the Gaussian integer \mathbb{G} using the inverse operator \mathcal{G}_ρ^{-1} . Indeed, for a given $x \in \mathbb{Z}[\rho]$ point from $\mathcal{S}_{q_1, q_2}^\rho$, the decoder \mathcal{D}_{q_1, q_2} is also defined as

$$\begin{aligned} \hat{c} &:= \mathcal{D}_{q_1, q_2}(\mathcal{G}_\rho^{-1}(x)), \\ &:= \Re(\mathcal{G}_\rho^{-1}(x))q_1 + \Im(\mathcal{G}_\rho^{-1}(x))q_2, \quad q_1, q_2 \in \mathbb{Z} \end{aligned} \quad (14)$$

Algorithm 1 SumComp Coding: Digital Coding Over-the-Air

```

1: Input: Function  $f(s_1, \dots, s_K)$ ,  $\rho$ , modulation order  $q_1, q_1$ 
   and  $q_2$  encoding scheme
2: Output: output of function  $\hat{f}$ 
3: Initialize:
   CP shared  $\varphi_k$  functions,  $q_1, q_2$  and  $\rho$  with the
   nodes
   Node  $k$  computes  $\mathcal{E}$  and  $\mathcal{G}_\rho$  by  $q_1, q_2$  and  $\rho$ 
4: procedure SUMCOMP ( $s_1, \dots, s_K$ )
5:   for in parallel  $k \leftarrow 1, 2, \dots, K$  do
6:     Node  $k$  computes  $c_k = \varphi_k(s_k)$ 
7:     Node  $k$  computes constellation point  $x_k$  using (8)
8:     Node  $k$  transmit  $x_k$  simultaneously for OAC
9:   end for
10:  if CP received  $r$  from (1) then
11:    CP maps  $r$  to the nearest neighbor in  $\mathbb{Z}[\rho]$  to
   obtain  $\hat{x}$  using (13).
12:    CP computes  $\hat{c}_k$  using (14)
13:    CP obtains output by  $\hat{f} = \psi(\sum_k \hat{c}_k)$ 
14:  end if
15: end procedure

```

where $x = \mathcal{Q}_\rho(\mu)$ is the output of the quantizer \mathcal{Q}_ρ . Therefore, by defining the decoding block as operator $\mathcal{D} := \psi \mathcal{D}_{q_1, q_2} \mathcal{G}_\rho^{-1} \mathcal{Q}_\rho$ that acts on the received signal at the receiver, the overall decoding procedure can be expressed as follows:

$$\hat{f} = \mathcal{D}(r) := \psi \left(\mathcal{D}_{q_1, q_2} \left(\mathcal{G}_\rho^{-1} (\mathcal{Q}_\rho(r)) \right) \right). \quad (15)$$

Next, we verify that the pair of $(\mathcal{E}, \mathcal{D})$ accurately reproduces the original input. For the noiseless scenario, i.e., $r = \sum_{k=1}^K x_k$, we can check that $f = \mathcal{D}(\sum_{k=1}^K x_k)$. Specifically, we can write

$$\begin{aligned} \mathcal{D} \left(\sum_{k=1}^K x_k \right) &= \psi \left(\mathcal{D}_{q_1, q_2} \left(\mathcal{G}_\rho^{-1} \left(\sum_{k=1}^K x_k \right) \right) \right), \\ &= \psi \left(\mathcal{D}_{q_1, q_2} \left(\sum_{k=1}^K \mathcal{G}_\rho^{-1}(x_k) \right) \right), \\ &= \psi \left(\Re \left(\sum_{k=1}^K g_k \right) q_1 + \Im \left(\sum_{k=1}^K g_k \right) q_2 \right), \\ &= \psi \left(\sum_{k=1}^K \left(\Re(g_k) q_1 + \Im(g_k) q_2 \right) \right), \\ &= \psi \left(\sum_{k=1}^K c_k \right) = \psi \left(\sum_{k=1}^K \varphi_k(s_k) \right) = f, \quad (16) \end{aligned}$$

where the first equality comes from the fact that without noise, the quantizer operator \mathcal{Q} acts as an identity operator. Moreover, the second equality is because of the linearity of the operator \mathcal{G}_ρ . In the presence of noise, the decoded function obviously results in error and $\hat{f} \neq f$. In the next section, we analyze the impact of the noise on the SumComp coding in terms of mean square and absolute error for different functions with various hyperparameters as q_1, q_2 , and ρ . The overall encoding and decoding procedure for SumComp coding is presented in Algorithm 1.

IV. SUMCOMP CODING: PERFORMANCE ANALYSIS

In this section, we analyze the SumComp coding in terms of desired function recovery from \hat{f} . It is clear that the analysis depends strongly on the desired function and, thus, on the post-processing function ψ , which is generally nonlinear. A unified error analysis is, therefore, very complicated and has to be done separately for every individual f . Consequently, we first focus on the simple, but important, special case of the *arithmetic sum* with helpful results on the error. Then, we extend the results to a larger class of nomographic functions where the post-processing function ψ is a uniformly continuous function.

A. MSE and MAE Analysis

To analyze the computation error, we use the classical MSE and MAE, which are respectively defined as

$$\text{MSE}(\hat{f}) := \mathbb{E}\{|f - \hat{f}|^2\}, \quad (17)$$

$$\text{MAE}(\hat{f}) := \mathbb{E}\{|f - \hat{f}|\}, \quad (18)$$

where the expectation is calculated over the randomness of the noise, $\hat{f} = \mathcal{D}(r)$ is the estimated value of the summation function at the receiver, and f is the true value, i.e., $f = \psi \left(\sum_{k=1}^K \varphi_k(s_k) \right)$. For the case where the desired function is the arithmetic sum case, or equivalently, $\varphi(s_k) = s_k$ for $k \in [K]$ and $\psi(g) = g$. Then, we have the following result regarding the MSE.

Proposition 2: Consider a communication network with K nodes where the nodes use the SumComp encoder \mathcal{E} to compute the summation $f = \sum_k s_k$, where $s_k \in \mathbb{Z}_q$ over the noisy MAC with $\mathcal{CN}(0, \sigma^2)$. Assuming the induced constellation points by $\sum_k s_k$ has uniform distribution over $[0, nK - 1]$, the MSE of the computation errors of using the constellation points set \mathcal{X} with $\Lambda_q = [M_1] \times [M_2]$ for two positive integers $M_1, M_2 \in \mathbb{Z}^+$, is given by

$$\text{MSE}(\hat{f}) = q_1^2 \alpha_1 + q_2^2 \alpha_2, \quad (19)$$

where

$$\begin{aligned} \alpha_1 &= 2 \sum_{\ell_1=1}^{M_1-1} \left(2\ell_1 - 1 + \frac{3\ell_1(1-\ell_1)-1}{M_1} \right) Q \left(\frac{(2\ell_1-1)}{2\sigma} \right), \\ \alpha_2 &= 2 \sum_{\ell_2=1}^{M_2-1} \left(2\ell_2 - 1 + \frac{3\ell_2(1-\ell_2)-1}{M_2} \right) Q \left(\frac{(2\ell_2-1)|\rho|}{2\sigma} \right), \end{aligned}$$

in which q_1, q_2 , and ρ are coprime integers and complex numbers for the ring of integer $\mathbb{Z}[\rho]$, respectively. Moreover, $Q(x)$ is the Gaussian Q function, i.e.,

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt. \quad (20)$$

Proof: See Appendix B. \square

Proposition 2 delineates that MSE comprises two components, each influenced by the selection of q_1 or q_2 . Modifying q_1 or q_2 inversely affects the counterpart and alters one of the error components. Additionally, a reduction in q_1 or q_2 elevates α_1 or α_2 , correspondingly. The optimal selection of q_1 or q_2 , contingent upon the chosen ρ , can achieve minimal MSE.

Remark 6: Assuming a uniform distribution across the constellation points induced by $\sum_k s_k$, the MSE specified in (19) represents the expected error value. Furthermore, when $\sigma/Kq \ll 1$, where K denotes the number of nodes and q the modulation order, the MSE provided in (19) holds, regardless of the prior distribution of s_k or the specific constellation points.

Then, the obtained analytical expression of the MSE can be extended to a class of *uniformly continuous* functions [44] defined as follows.

Definition 3: A function $\psi : \mathbb{S} \mapsto \mathbb{R}$ is called uniformly continuous function if there exists a strictly increasing concave function $w_\psi : [0, \infty) \mapsto [0, \infty)$ with $w_\psi(0) = 0$ such that, for all real $x, y \in \mathbb{S}$, we have

$$|\psi(x) - \psi(y)| \leq w_\psi(|x - y|), \quad (21)$$

where w_ψ is called the modulus of continuity of ψ .

Some examples of these functions are listed below:

- For L -Lipschitz function $\psi : \mathbb{R} \mapsto \mathbb{R}$, the modulus function becomes $w_\psi(x) = Lx$.
- If ψ is a Hölder continuous with (C, α) for $\alpha \in (0, 1]$, i.e., for all x, y in the domain of ψ , we have $|\psi(x) - \psi(y)| \leq C|x - y|^\alpha$, then w_ψ reads to $x \mapsto Cx^\alpha$.
- When ψ is an increasing concave function, then w_ψ becomes trivial, i.e., $w_\psi : x \mapsto x$.

In the following, we have an upper bound on the MAE for the nomographic function with uniformly continuous post-processing function ψ .

Proposition 3: Consider a communication network with K nodes, where node k uses the encoder \mathcal{E}_k to compute the summation $f = \psi\left(\sum_{k=1}^K \varphi_k(s_k)\right)$, and ψ is an uniformly continuous function over the noisy MAC with $\mathcal{CN}(0, \sigma^2)$. Also, assume the induced constellation points by $\sum_k s_k$ have uniform distribution over $[0, nK - 1]$. Then, the MAE of the computation errors when utilizing the constellation points from the set \mathcal{X} with Λ_q , as in Proposition 2, is confined within the following upper bound

$$\text{MAE}(\hat{f}) \leq w_\psi(q_1\beta_1 + q_2\beta_2), \quad (22)$$

where w_ψ signifies the modulus of continuity of ψ , and β_1 and β_2 are given by

$$\beta_1 = 2 \sum_{\ell_1=1}^{M_1-1} \left(1 + \frac{1-2\ell_1}{M_1}\right) Q\left(\frac{(2\ell_1-1)}{2\sigma}\right), \quad (23a)$$

$$\beta_2 = 2 \sum_{\ell_2=1}^{M_2-1} \left(1 + \frac{1-2\ell_2}{M_2}\right) Q\left(\frac{(2\ell_2-1)|\rho|}{2\sigma}\right). \quad (23b)$$

Proof: See Appendix C. \square

Different from Proposition 2, Proposition 3 establishes that the MAE of \hat{f} is subject to an upper limit rather than an average value. Nevertheless, this upper bound elucidates a compromise between the selections of q_1 and q_2 . The relationship of this upper bound with q_1 and q_2 varies according to the chosen w_ψ function, potentially exhibiting linear, quadratic, or other forms of dependency. This differs from the quadratic relationship identified in the MSE scenario.

Corollary 1: Consider that ψ is Hölder continuous with (C, α) . Then, for any nomographic function f with post-processing $\psi(\cdot)$, the MAE is upper bounded by

$$\text{MAE}(\hat{f}) \leq C|q_1\beta_1 + q_2\beta_2|^\alpha, \quad (24)$$

where $\alpha \in (0, 1]$.

To illustrate, we take a few examples here.

Example 3: (Maximum) The maximum function $f = \max_k s_k$ can be approximated with $\varphi_k(s_k) = \exp(s_k)$ and $\psi(x) = \ln(x)$. Then, $\psi(x)$ is a concave and strictly increasing function. Therefore, w_ψ is the identity function. However, if we bound the min of x by a factor θ , i.e., $\theta \leq x$, then the logarithmic function becomes Lipschitz continuous with $1/\theta$. Therefore, we have

$$\text{MAE}(\hat{f}) \leq \frac{q_1\beta_1 + q_2\beta_2}{\theta}. \quad (25)$$

Example 4: (Euclidean norm) Let the desired function be the Euclidean norm, i.e., $f = \sqrt{s_1^2 + \dots + s_K^2}$ where $\varphi_k(s_k) = s_k^2$ and $\psi(x) = \sqrt{x}$. Hence, ψ is Hölder continuous with $\alpha = 0.5$, and consequently, it yields

$$\text{MAE}(\hat{f}) \leq \sqrt{|q_1\beta_1 + q_2\beta_2|}. \quad (26)$$

Example 5: (Arithmetic mean) Let function f be the arithmetic mean function, i.e., $\sum_{k=1}^K s_k / K$. Then $\varphi_k(s_k) = s_k$ and $\psi(x) = x/K$ which is a concave and strictly increasing function. As a result, w_ψ is $1/K$, i.e.,

$$\text{MAE}(\hat{f}) \leq \frac{q_1\beta_1 + q_2\beta_2}{K}. \quad (27)$$

B. Complexity Analysis

In this section, we compare the complexity of SumComp coding to the coding scheme proposed in ChannelComp in terms of the number of basic operators (BOPs) to show the obtained reduction in the cost. Indeed, ChannelComp [32] uses a coding that is based on an optimization problem where the input is a vector involving the constellation points of the modulation diagram of all the nodes to obtain the encoder \mathcal{E}_k for all the nodes. Also, the decoder \mathcal{D} is then determined as a tabular function that maps the resultant constellations to the output co-domain of the function f .

In the sequel, we analyze the complexity of the encoder and decoder. Consider that the input vector is of size $q \times K$, where q denotes the quantization levels and K is the number of nodes in the network. The complexity of solving a semidefinite programming optimization in ChannelComp is $\mathcal{O}(\max\{n, m\}^4 n^{0.5})$ [45], [46], in which n is the dimension of the input variable, i.e., $q \times K$, and m is the number of constraints, which is at most $m \leq q^K(1 + q^K)/2$. As a result, the computational complexity to find a valid modulation vector in ChannelComp is at most $\mathcal{O}(q^{0.5} K^{0.5} q^{8K})$, or simply $\mathcal{O}(\sqrt{K} q^{8K+0.5})$, for computing a general function. Moreover, for the case of symmetric functions, the modulation vectors are the same for all K nodes, which implies that $n = q$. Further, the number of constraints becomes at most $m \leq \binom{K+q-1}{q-1} < \exp(K + q - 1)$ [32]. Therefore, the overall computational complexity reduces to $\mathcal{O}(\sqrt{q} e^{4(K+q-1)})$.

For analyzing the complexity of the decoder part, we need to know the number of constellation points, which depends on the output of the optimization problem and can not be determined beforehand. Hence, it is difficult to consider an exact number. However, the number of constellation points can reach at most q^K different values. In this case, using a linear search for the tabular function, ChannelComp can decode the received signal at most in $\mathcal{O}(q^K)$ BOPs [47].

However, SumComp coding offers an optimization-free approach. To analyze the complexity of the encoder \mathcal{E}_k including the composite operator $\mathcal{G}_\rho \mathcal{E} \varphi_k$, we need to analyze each operator separately. Similarly, we should consider a similar approach for the decoder \mathcal{D} with its sub-operators. The following proposition establishes the complexity of a class of nomographic functions.

Proposition 4: For a given function $f = \psi\left(\sum_{k=1}^K \varphi_k(s_k)\right)$, where $\varphi_k : \mathbb{R} \mapsto [-a, a]$ and $\psi : [-a, a] \mapsto [-b, b]$. Also, consider that the m_k -th derivation of φ_k and ℓ -th derivation of ψ exists and are bounded by factors E_k and D , respectively. The complexity of the encoders for computing the desired function f over a network with K nodes and the CP server is given by

$$\#\text{BOPs} = \mathcal{O}\left(\sum_{k=1}^K \frac{\ln\left(\frac{E_k q}{2a\sqrt{2\pi}}\right)}{\mathcal{W}\left((2ae)^{-1} \ln\left(\frac{E_k q}{2a\sqrt{2\pi}}\right)\right)}\right), \quad (28)$$

and for the decoder by

$$\#\text{BOPs} = \mathcal{O}\left(\frac{\ln\left(\frac{Dq}{2a\sqrt{2\pi}}\right)}{\mathcal{W}\left((2be)^{-1} \ln\left(\frac{Dq}{2a\sqrt{2\pi}}\right)\right)}\right), \quad (29)$$

for $q \gg 1$ and $\mathcal{W}(\cdot)$ denotes the Lambert \mathcal{W} function.

Proof: See Appendix D. □

From Proposition 4, we note that the computational complexity of the encoder and the decoder of the SumComp coding are approximately $\mathcal{O}(K \ln(q))$ and $\mathcal{O}(\ln(q))$, respectively.

Therefore, given the complexity of the SumComp coding and the original coding in ChannelComp, we can conclude that the SumComp coding has an order of magnitude less computation costs than the coding included in ChannelComp, specifically when $q \gg 1$.

In the next section, we evaluate the performance of the SumComp coding for computing different functions and compare it with the other methods, such as AirComp and the coding in ChannelComp.

V. NUMERICAL RESULT

In this section, we assess the performance of the SumComp coding under various digital modulation schemes and corroborate the theoretical analysis through numerical results. In particular, we first evaluate the performance of SumComp coding with standard digital modulations, such as QAM and PAM, with different orders for computing summation function in terms of MSE metric. Moreover, we compare the output of empirical results from the simulation with the theoretical results from Section IV. We repeat this comparison for computing various functions in the class of nomographic functions in terms of the MAE metric. Afterward, we assess

the performance of SumComp coding compared to the other existing methods. More precisely, the performance of the SumComp coding is compared to three different methods:

- ChannelComp original encoding: the optimization-based coding scheme proposed in [32] uses an optimization problem to obtain the digital modulation vectors.
- AirComp method: the traditional AirComp method [4], [18], where nodes use analog modulation for communication.
- Standard digital transmission: this scenario implements the computation method that relies on the naive OFDMA. This method ensures that each node is allocated unique frequency channels and prevents potential communication interference with more bandwidth consumption.

Note that our comparison is limited to state-of-the-art coherent AirComp methods. Non-coherent methods, while viable, demand significantly higher communication resources compared to coherent counterparts, due to the absence of channel compensation.

Finally, we analyze the computational complexity of SumComp coding in comparison with the optimization-based coding in ChannelComp in terms of the number of BOPs. Our exploration and comparison over different metrics aim at providing a comprehensive understanding of the potential utility and performance of the SumComp coding. Note that we use times sign, star, triangle, square, and pentagon markers to show the analytical results, plus signs, reverse triangles, diamonds, reverse pentagons, and circle markers for the empirical results. Also, warm colors are reserved for QAM modulation, and cool colors are used for PAM modulation. All the numerical experiment codes are provided on Github.³

A. SumComp Coding With QAM & PAM Modulations

First, we analyze the performance of \mathcal{E}_k encoders and \mathcal{D} decoder for computing the summation function $f = \sum_{k=1}^K s_k$ over the MAC with $K = 100$ nodes. The input data are generated in the same order as the modulations, i.e., $s_k \in \mathbb{Z}_q$ for $k \in [K]$, where q is the order of modulation. We consider two cases: nodes use QAM with $q = \{16, 64, 256\}$; and PAM with $q = \{16, 32, 64\}$ over Gaussian channel for different levels of SNR, which is defined as $\text{SNR} := 10 \log(\sum_{i \in \mathcal{X}} |x_i|^2 / q\sigma^2)$. We consider 5×10^4 Monte Carlo trials.

In Figure 6, we show the performance of the SumComp coding in terms of the MSE error when employed in different orders of QAM and PAM modulations. We also depict the analytical value of MSE in Proposition 2 for comparison with the outage of empirical error values. Figure 6(a) shows the empirical MSE error of the QAM of order q where $q \in \{16, 64, 256\}$ with the triangle, pentagon, and circle markers, respectively. Similarly, we show the empirical MSE for the PAM of order q , where $q \in \{16, 32, 64\}$ with the plus sign, triangle, and square markers, respectively in Figure 6(b). We note that the analytical MSE predicts well the average empirical error, as depicted in Figure 6. Moreover, by increasing the order of the modulations, the MSE values are higher over all ranges of SNR, which is expected because higher possible

³<https://github.com/SaeedRazavikia/SumComp-Coding-for-Digital-OAC.git>

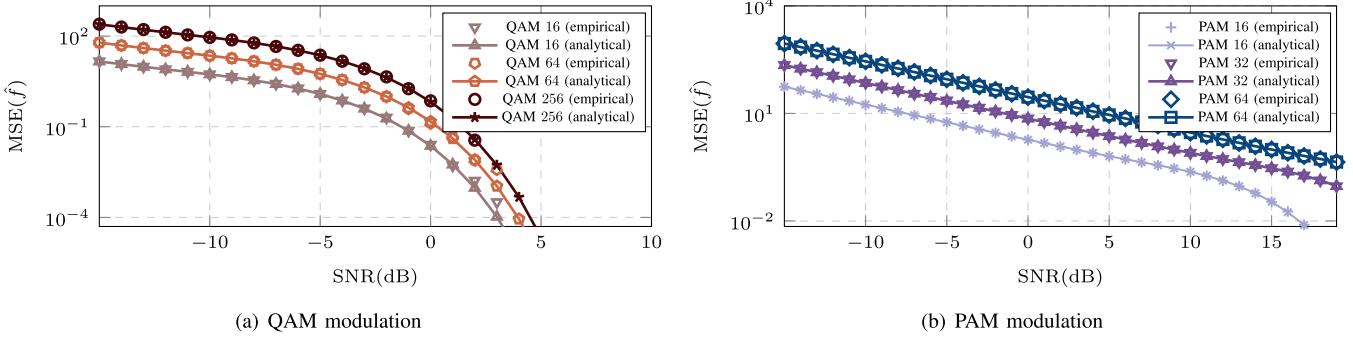


Fig. 6. Monte Carlo numerical evaluation of the MSE of computing the summation function for 5×10^4 trials versus the analytical results from Proposition 2 for digital modulation. Figures 6(a) and 6(b) show both empirical and analytical MSE for QAM modulations of order $q = \{16, 64, 256\}$, and PAM modulation of order $q = \{16, 32, 64\}$, respectively.

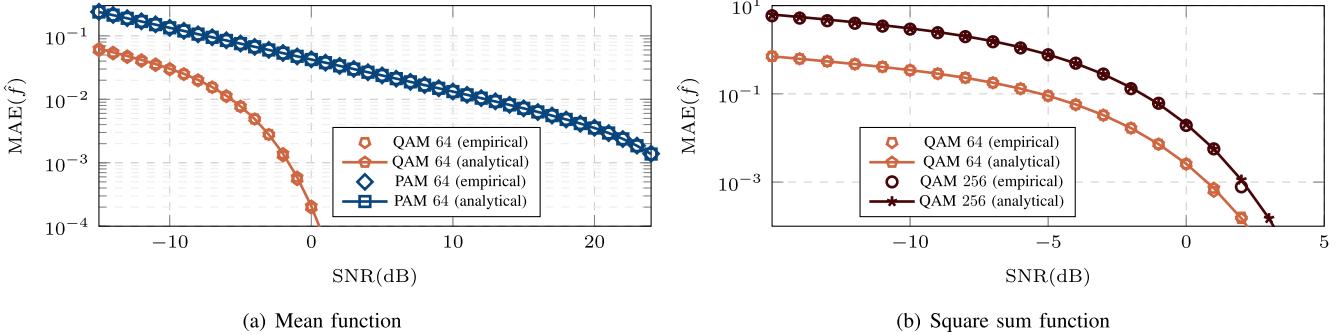


Fig. 7. Monte Carlo evaluation of MAE of computing the mean and square norm function for 5×10^4 trials versus analytical results (Proposition 3) for digital modulation. Figure 7(a) shows both empirical and analytical MAE for QAM modulations and PAM modulation both of order $q = 64$ in computing arithmetic mean function. Also, Figure 7(b) shows both empirical and analytical MAE for QAM modulations of order $q = \{64, 256\}$ for computing the norm function.

error values can occur due to the large size of the input field \mathbb{Z}_q . This observation is also consistent with the analytical value of the MSE in Proposition 2.

B. SumComp Coding for Nomographic Function

We consider nomographic functions and check the performance of the SumComp coding for two cases: the arithmetic mean and square sum functions (square Euclidean norm), as given in Examples 5 and 4, respectively. The computation is performed over a Gaussian MAC for $K = 100$ nodes. In the case of arithmetic mean, the input data of node k , i.e., s_k is uniformly random generated from \mathbb{Z}_q , where q is set to be 64 for both PAM and QAM modulations. Then, in the next case, we repeat the experiment with QAM modulation with low and high order modulation, i.e., $q = 64$ and 256. Moreover, the input data of node k , s_k , is generated uniformly random from $\{1, \dots, \sqrt{q}\}$ to make $c_k = \varphi(s_k) \in \mathbb{Z}_q$.

Figure 7 shows the performance of the SumComp coding for computing the arithmetic mean and the square sum functions in terms of MAE for different SNRs values. In Figure 7(a), the MAE of computing arithmetic mean functions are depicted for different SNRs for nodes using either QAM or PAM modulations. We observe that the proposed upper bound in Proposition 3 approximates well the behavior of MAE curves, and the empirical results are consistent with Proposition 3. Notably, QAM outperforms the PAM thanks to its power-efficient constellation diagram. Indeed, the mean energy per transmitted symbol of PAM and QAM of order q are $(q^2 - 1)A_s^2/12$ and $(q - 1)A_s^2/6$, respectively, where A_s denotes the distance between the constellation points. Thus, the constellation diagram of QAM can approximately provide

$10 \log(q + 1)/2$ SNR gain for the estimation of the received signal compared to PAM.

Similarly, Figure 7(b) shows the MAE error for computing the square sum function where nodes use QAM modulation with order $q = 64$ and $= 256$. Similar to Figure 6, increasing the order of modulation results in a higher MAE error. Similar to the arithmetic mean, the empirical results of MAE curves for the squared sum function are consistent with the upper bound proposed in Proposition 3.

C. SumComp Coding Vs ChannelComp, AirComp, and OFDMA

Now, we compare the SumComp coding with three methods: ChannelComp, AirComp, and OFDMA methods. The functions to compute are the arithmetic mean and geometric mean in terms of the normalized MSE (NMSE) metric, which is defined as $\text{NMSE} := \sum_{j=1}^{N_s} |f_j - \hat{f}_j|^2 / N_s |f_j|$, where N_s denotes the number of Monte Carlo trials, f_j denotes the value of the desired function we wish to compute, and \hat{f}_j is the estimated value of f_j for $j \in [N_s]$.

Figure 8(a) shows the NMSE error for computing the arithmetic mean function. We note that SumComp coding outperforms all the other methods for computing the summation function even at the low SNR regime (less than 0 dB). Specifically, for SNR around -5 dB, SumComp coding shows at least a 15 dB improvement compared to the other methods. This significant improvement in the performance of SumComp coding is mainly attributed to the power-efficient constellation diagram of QAM compared to PAM. We note that the apparent similarity in performance between ChannelComp and AirComp in this figure is noteworthy. This resemblance arises

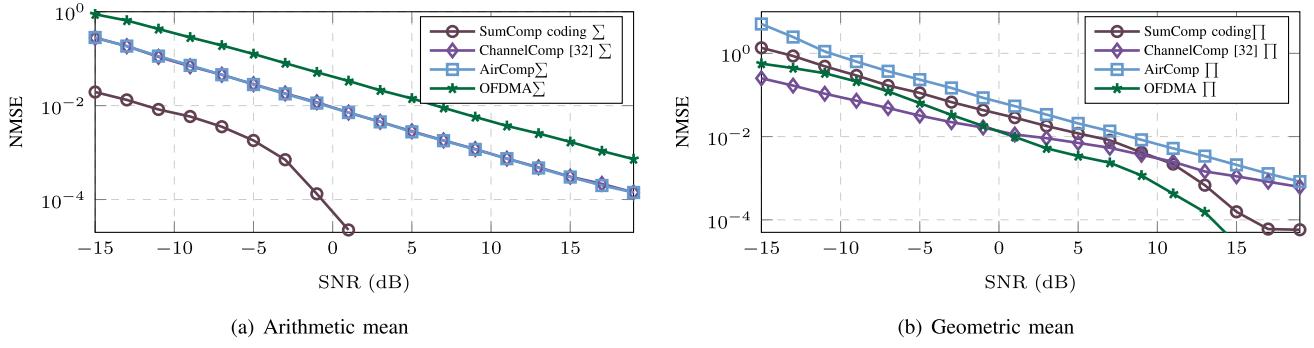


Fig. 8. Performance comparison between SumComp coding, coding in ChannelComp, AirComp, and OFDMA in terms of NMSE averaged over $N_s = 5 \times 10^4$ Monte Carlo trials, when values of the function to be computed are originally quantized. The input values are set to $s_k = \{1, 2, \dots, 64\}$ for the arithmetic mean, and $s_k = \{1, \dots, 8\}$ for the geometric mean functions. Specifically, the desired functions are $f_1 = \sum_{k=1}^{10} s_k / K, f_2 = (\prod_{k=1}^4 s_k)^{\frac{1}{K}}$.

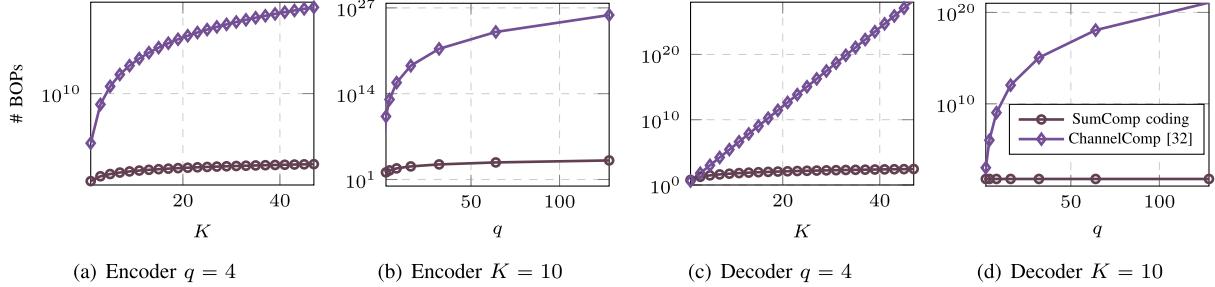


Fig. 9. Computational complexity for both encoder and decoder of SumComp coding and coding in ChannelComp in terms of BOPs. Figures 9(a) and 9(c) show the encoder's and decoder's complexity when $q = 4$ for different numbers of node K , respectively. Similarly, Figures 9(b) and 9(d) show the encoder's and decoder's complexity when $K = 10$ for different quantization levels q , respectively.

from the specific characteristics of the summation function used in these scenarios. In the case of ChannelComp, the scheme employs the PAM modulation, while AirComp utilizes the analog AM modulation. However, when the input data for AirComp is discretized, the analog AM modulation effectively becomes PAM, thus aligning it with the ChannelComp method.

We repeat this experiment for the geometric mean in Figure 8(b). We observe that SumComp coding exhibits superior performance compared to ChannelComp and AirComp in the high SNR regime. In particular, for SNR greater than 12 dB, SumComp coding NMSE has superiority over ChannelComp and AirComp, and it keeps increasing the performance for higher SNR until we observe more than 10 dB improvement at SNRs around 19 dB. Moreover, ChannelComp and OFDMA work better than the other methods for the low SNR regime.

Therefore, the results show that SumComp coding performs better in NMSE than other methods from the literature in computing the arithmetic and geometric mean, owing to the beneficial intersection of all nodes' constellation points and power-efficient QAM modulation. Additionally, the results imply that SumComp coding exceeds the conventional AirComp approach across a broad range of SNR values, exhibiting notable superiority, particularly in the high SNR regime compared to both ChannelComp and AirComp.

D. Complexity Performance

Finally, we compare the computational complexity of the SumComp coding with the optimization-based method by ChannelComp [8].

We show the order of complexity of ChannelComp and SumComp coding in Figure 9 for the summation function,

where the number of constraints is reduced to $K^2 q^2$, and because of symmetry, the dimension of variables becomes $n = q$. Therefore, the overall computational complexity for both the encoder and decoder of the summation function becomes $\mathcal{O}(K^8 q^{8.5})$. For SumComp coding, the required number of operations for the encoder is simply $\mathcal{O}(K)$, and for computing, the decoder is $\mathcal{O}(1)$ because $\psi(y) = y$. In Figure 9, the complexity of ChannelComp and SumComp coding are depicted for different numbers of nodes and quantization levels, respectively. We observe that the SumComp coding can significantly reduce the huge complexity of the optimization solution for ChannelComp, while it shows similar performance in the computing class of nomographic functions.

VI. CONCLUSION

In conclusion, this study identified the potential to enhance the ChannelComp coding procedure by merging communication and computation, but the inherent complexity poses challenges concerning its broad optimization applicability. To mitigate these challenges, we introduced the SumComp coding, an innovative and straightforward coding scheme based on the ring of integers that allows for the implementation of digital modulations. The SumComp coding exhibited remarkable compatibility across several digital modulations, such as QAM and PAM, to name a few. In addition, we analyzed the MSE for SumComp coding in the computation of the arithmetic mean function and established an upper bound on the MAE for various nomographic functions.

An important conclusion from our work is the demonstrable superiority of SumComp coding over traditional AirComp methods. The empirical results substantiated this claim and

implied significant advancement in digital modulation for Air-Comp. Specifically, for computing arithmetic and geometric mean in numerical results, SumComp coding shows around 10 dB improvements in terms of normalized MSE for low-noise scenarios. The proposed methodology of SumComp coding not only overcomes the computational complexity of the original coding scheme of ChannelComp but also paves the way for further exploration of digital wireless computation.

The potential of SumComp coding inspires many avenues for further study. Here, we summarize the most promising trajectories.

- **Channel code enhancement:** develop a SumComp channel code resistant to noise, such that optimizing the transmission number and computation error becomes possible.
- **Expanding function computation:** while SumComp coding successfully computes summation functions, considering other ring algebraic structures, such as the polynomial ring and the quotient ring, could allow for a wider range of functions computed over the air.
- **MIMO extension for matrix computation:** progressing from a single narrowband antenna system to a MIMO system enables vector-based computations, paving the way for more applications, such as matrix computation and federated learning.
- **Machine learning applications:** we envision that Sum-Comp coding can be applied and significantly enhance applications in edge federated learning.
- **Real-world implementation:** evaluating the SumComp coding with real-world experiments to demonstrate Sum-Comp's feasibility in practical wireless communication systems.

APPENDIX

A. Proof of Proposition 1

For $\mathcal{S}_{q_1, q_2}^\rho$ to be a group under the operation of summation, it needs to satisfy the *closure*, *associativity*, *identity element*, and *inverse element* property of a group. Each property can be proved as follows. First, note that $\mathcal{S}_{q_1, q_2}^\rho$ is non-empty because it contains at least the identity element $(0, 0)$.

Closure: Let us suppose there are two elements (c_1, x_1) and (c_2, x_2) in $\mathcal{S}_{q_1, q_2}^\rho$, where $c_1, c_2 \in \mathbb{Z}$, and $x_1, x_2 \in \mathbb{Z}[\rho]$. Then, we need to prove that their sum, which is $(c_1 + c_2, x_1 + x_2)$, also belongs to $\mathcal{S}_{q_1, q_2}^\rho$. To this end, we have

$$c_1 = \Re(\mathcal{G}_\rho^{-1}(x_1))q_1 + \Im(\mathcal{G}_\rho^{-1}(x_1))q_2, \quad (30)$$

$$c_2 = \Re(\mathcal{G}_\rho^{-1}(x_2))q_1 + \Im(\mathcal{G}_\rho^{-1}(x_2))q_2. \quad (31)$$

By adding up both sides, we have

$$\begin{aligned} c_1 + c_2 &= \Re(\mathcal{G}_\rho^{-1}(x_1 + x_2))q_1 \\ &\quad + \Im(\mathcal{G}_\rho^{-1}(x_1 + x_2))q_2 \in \mathcal{S}_{q_1, q_2}^\rho. \end{aligned} \quad (32)$$

Therefore, $(c_1 + c_2, x_1 + x_2) \in \mathcal{S}_{q_1, q_2}^\rho$ and the closure property is satisfied. *Associativity:* This property holds because the operation of addition is associative in \mathbb{R}^2 and \mathbb{R} .

Identity element: The identity element is $(0, 0)$ since for any element $(c, x) \in \mathcal{S}_{q_1, q_2}^\rho$, we have $(0 + c, 0 + x) = (c, x)$.

Inverse element: For every element $(c, x) \in \mathcal{S}_{q_1, q_2}^\rho$, the inverse is given by $(-c, -x)$ because $c + (-c) = 0$ and $x + (-x) = 0$.

Since all group axioms are satisfied, we conclude that $\mathcal{S}_{q_1, q_2}^\rho$ is a group under the operation of summation.

B. Proof of Proposition 2

To compute the MSE for the arithmetic sum function, i.e., $f = \sum_{k=1}^K s_k$ for $s_k \in \mathbb{Z}_q$, we first need to compute the probability of the error for each individual value of constellation points. To this end, we have

$$\begin{aligned} r &= \sum_{k=1}^K x_k + z = \sum_{k=1}^K \{\mathcal{G}_\rho(\mathcal{E}_{q_1, q_2}(s_k))\} + z, \\ &= \sum_{k=1}^K \{s_k \mu_1 + mq_2 + (s_k \mu_2 - mq_1)i\rho\} + z, \\ &= \sum_{k=1}^K \{s_k \mu_1 + mq_2 + (s_k \mu_2 - mq_1)i\rho\} + z_1 + z_2 \rho i, \end{aligned} \quad (33)$$

where $\mu_1 q_1 + \mu_2 q_2 = 1$, and $z_1 + z_2 \rho \in \mathbb{C}$ is the representation of noise z in the ring of $\mathbb{Z}[\rho]$. Then, using (33), the decoder \mathcal{D} yields

$$\begin{aligned} \hat{f} &= \mathcal{D}_{q_1, q_2} \mathcal{G}_\rho^{-1} \mathcal{Q}_\rho(r) \\ &= \mathcal{D}_{q_1, q_2} \mathcal{G}_\rho^{-1} \left(\sum_{k=1}^K \{s_k \mu_1 + mq_2\} + \mathcal{Q}_\rho(z_1) \right. \\ &\quad \left. + \sum_{k=1}^K \{(s_k \mu_2 - mq_1)i\rho + \mathcal{Q}_\rho(z_2)\rho i\} \right), \\ &= \mathcal{D}_{q_1, q_2} \left(\sum_{k=1}^K \{s_k \mu_1 + mq_2\} + \mathcal{Q}_\rho(z_1) \right. \\ &\quad \left. + \sum_{k=1}^K \{(s_k \mu_2 - mq_1)i + \mathcal{Q}_\rho(z_2)i\} \right), \\ &= q_1 \times \sum_{k=1}^K \{s_k \mu_1 + mq_2\} + q_1 \times \mathcal{Q}_\rho(z_1) + \\ &\quad q_2 \times \sum_{k=1}^K \{(s_k \mu_2 - mq_1) + q_2 \times \mathcal{Q}_\rho(z_2)\}, \\ &= \sum_{k=1}^K \{s_k \mu_1 q_1 + mq_2 q_1\} + \sum_{k=1}^K \{s_k q_2 \mu_2 - mq_1 q_2\} \\ &\quad + q_1 \mathcal{Q}_\rho(z_1) + q_2 \mathcal{Q}_\rho(z_2), \\ &= f + q_1 \mathcal{Q}_\rho(z_1) + q_2 \mathcal{Q}_\rho(z_2). \end{aligned} \quad (34)$$

Hence, for the MSE, we can write

$$\begin{aligned} \mathbb{E}\{|f - \hat{f}|_2^2\} &= \mathbb{E}\{|q_1 \mathcal{Q}_\rho(z_1) + q_2 \mathcal{Q}_\rho(z_2)|^2\}, \\ &= q_1^2 \mathbb{E}\{\underbrace{|\mathcal{Q}_\rho(z_1)|^2}_{:=e_1}\} + q_2^2 \mathbb{E}\{\underbrace{|\mathcal{Q}_\rho(z_2)|^2}_{:=e_2}\}, \end{aligned} \quad (35)$$

where the last equality comes from the fact that z_1 and z_2 are statistically independent random variables. Since noise components z_1 and z_2 are independent random variables, we can independently treat the error of the real part e_1 and the imaginary part e_2 in (34). The operator \mathcal{Q}_ρ in 1D acts as the round function, and it maps the input values from \mathbb{R} to

its nearest neighbor integer value. Consequently, variables e_1 and e_2 are integers corresponding to the modulation's error in-phase and quadrature components. Because of noise in every dimension, the receiver may add the discretized value of error z_1 (or z_2) from the \mathbb{Z} to the estimated value \hat{f} . Then, the computation error in the real domain equals the absolute value of e_1 times the step size q_1 . Similarly, the computation error in the imaginary domain is $|e_2| \times q_2$.

In this regard, let P_1^ℓ and P_2^ℓ be the probability that $e_1 = \ell$ and $e_2 = \ell$, respectively. Then, we have from [48] and using the law of total expectation:

$$\mathbb{E}[e_1^2] = \sum_{j_1=0}^{M_1-1} \mathbb{E}[e_1^2 | s = j_1] \Pr[s = j_1], \quad (36a)$$

$$\mathbb{E}[e_2^2] = \sum_{j_2=0}^{M_2-1} \mathbb{E}[e_2^2 | s = j_2] \Pr[s = j_2], \quad (36b)$$

where $s = \sum_{k=1}^K s_k$. Under the assumption of a uniform distribution over the induced constellation points, i.e., $\Pr[s = j_1] = 1/M_1$ and $\Pr[s = j_2] = 1/M_2$,⁴ Then for the conditional probability $\mathbb{E}[e_1 | s = j]$ and $\mathbb{E}[e_1 | s = j]$, we have

$$\mathbb{E}[e_1^2 | s = j_1] = \sum_{\ell_1=1}^{M_1-j_1-1} \ell_1^2 \zeta_1(\ell_1), \quad (37a)$$

$$\mathbb{E}[e_2^2 | s = j_2] = \sum_{\ell_2=1}^{M_2-j_2-1} \ell_2^2 \zeta_2(\ell_2), \quad (37b)$$

where ζ_1 and ζ_2 functions are defined as

$$\zeta_1(\ell_1) := Q\left(\frac{(2|\ell_1|-1)d_1}{\sigma_1}\right) - Q\left(\frac{(2|\ell_1|+1)d_1}{\sigma_1}\right), \quad (38a)$$

$$\zeta_2(\ell_2) := Q\left(\frac{(2|\ell_2|-1)d_2}{\sigma_2}\right) - Q\left(\frac{(2|\ell_2|+1)d_2}{\sigma_2}\right), \quad (38b)$$

for $\ell_1 = 1 - M_1, \dots, -1, 0, 1, \dots, M_1 - 1$ and $\ell_2 = 1 - M_2, \dots, -1, 0, 1, \dots, M_2 - 1$, where d_1 and d_2 are the in-phase and quadrature decision distances, respectively, which are $d_1 = d_2 = 1/2$ for Gaussian integers. Moreover, σ_1 and σ_2 denote the variance of noise z_1 and z_2 , which are $\sigma_1 = \sigma$ and $\sigma_2 = \sigma/|\rho|$, respectively. Also, recall that $Q(x)$ is the Gaussian Q function.

Then, by substituting (37) into (36), the expected value $\mathbb{E}[e_1^2]$ can be computed as

$$\mathbb{E}[e_1^2] = \frac{1}{M_1} \sum_{j_1=0}^{M_1-1} \sum_{\ell_1=1}^{M_1-j_1-1} \ell_1^2 \zeta_1(\ell_1),$$

By changing the order of summations, we obtain

$$\begin{aligned} \mathbb{E}[e_1^2] &= \frac{1}{M_1} \sum_{\ell_1=1}^{M_1-1} \sum_{j_1=0}^{M_1-\ell_1-1} \ell_1^2 \times \zeta_1(\ell_1), \\ &= \frac{1}{M_1} \sum_{\ell_1=1}^{M_1-1} (M_1 - \ell_1) \ell_1^2 \zeta_1(\ell_1), \\ &= \sum_{\ell_1=1}^{M_1-1} \left(1 - \frac{\ell_1}{M_1}\right) \ell_1^2 \zeta_1(\ell_1), \\ &= \sum_{\ell_1=1}^{M_1-1} \left(\ell_1^2 - \frac{\ell_1^3}{M_1}\right) \zeta_1(\ell_1). \end{aligned} \quad (39)$$

By following a similar procedure, $\mathbb{E}[e_2^2]$ can be computed as

$$\mathbb{E}[e_2^2] = \sum_{\ell_2=1}^{M_2-1} \left(\ell_2^2 - \frac{\ell_2^3}{M_1}\right) \zeta_2(\ell_2). \quad (40)$$

⁴This is the marginal probability distribution for both the real and the imaginary axes. Since the joint distribution is uniform, then its marginal also becomes uniform.

Next, we eliminate the subsequent terms in both sums in (39) and (40) to simplify the expression; we obtain the following expression for the error terms.

$$\mathbb{E}[e_1^2] = \sum_{\ell_1=1}^{M_1-1} \frac{\alpha_1(\ell_1)}{q_1^2} Q\left(\frac{2\ell_1-1}{2\sigma}\right), \quad (41a)$$

$$\mathbb{E}[e_2^2] = \sum_{\ell_2=1}^{M_2-1} \frac{\alpha_2(\ell_2)}{q_2^2} Q\left(\frac{(2\ell_2-1)|\rho|}{2\sigma}\right), \quad (41b)$$

in which,

$$\alpha_1(\ell_1) = 2q_1^2 \left(2\ell_1 - 1 + \frac{3\ell_1(1-\ell_1)-1}{M_1}\right), \quad (42a)$$

$$\alpha_2(\ell_2) = 2q_2^2 \left(2\ell_2 - 1 + \frac{3\ell_2(1-\ell_2)-1}{M_2}\right). \quad (42b)$$

Therefore, we conclude the proof of Proposition 2.

C. Proof of Proposition 3

By recalling the definition of the MSE(\hat{f}) = $\mathbb{E}[|f - \hat{f}|]$, we have

$$\begin{aligned} \mathbb{E}[|f - \hat{f}|] &= \mathbb{E}\left[\left|\psi\left(\sum_{k=1}^K s_k\right) - \psi\left(\sum_{k=1}^K \hat{s}_k\right)\right|\right], \\ &\leq \mathbb{E}\left[\left|w_\psi\left(\left|\sum_{k=1}^K s_k - \sum_{k=1}^K \hat{s}_k\right|\right)\right|\right], \\ &= \mathbb{E}\left[w_\psi\left(\left|\sum_{k=1}^K s_k - \sum_{k=1}^K \hat{s}_k\right|\right)\right], \\ &\leq w_\psi\left(\mathbb{E}\left[\left|\sum_{k=1}^K s_k - \sum_{k=1}^K \hat{s}_k\right|\right]\right), \end{aligned} \quad (43)$$

where the final inequality represents the reversed Jensen's inequality [49], due to the concavity of w_ψ . To obtain the expected value in the argument of function w_ψ , we use (34) as follows

$$\begin{aligned} \mathbb{E}\left[\left|\sum_{k=1}^K s_k - \sum_{k=1}^K \hat{s}_k\right|\right] &= \mathbb{E}\{|q_1 \mathcal{Q}_\rho(z_1) + q_2 \mathcal{Q}_\rho(z_2)|\}, \\ &\leq q_1 \mathbb{E}\{|Q_\rho(z_1)|\} + q_2 \mathbb{E}\{|Q_\rho(z_2)|\}, \end{aligned} \quad (44)$$

where the last inequality is due to the triangle inequality. By following a similar procedure from (36) to (39), we can compute the expectation of the error terms as follows:

$$\mathbb{E}[e_1] = \sum_{\ell_1=1}^{M_1-1} \left(\ell_1 - \frac{\ell_1^2}{M_1}\right) \zeta_1(\ell_1), \quad (45a)$$

$$\mathbb{E}[e_2] = \sum_{\ell_2=1}^{M_2-1} \left(\ell_2 - \frac{\ell_2^2}{M_2}\right) \zeta_2(\ell_2). \quad (45b)$$

With further simplifications, we obtain the following terms:

$$\mathbb{E}[e_1] = 2 \sum_{\ell_1=1}^{M_1-1} \left(1 + \frac{1-2\ell_1}{M_1}\right) Q\left(\frac{(2\ell_1-1)}{2\sigma}\right), \quad (46a)$$

$$\mathbb{E}[e_2] = 2 \sum_{\ell_2=1}^{M_2-1} \left(1 + \frac{1-2\ell_2}{M_2}\right) Q\left(\frac{(2\ell_2-1)|\rho|}{2\sigma}\right). \quad (46b)$$

Then, by substituting (46) into (44), we obtain

$$\mathbb{E}\left[\left|\sum_{k=1}^K s_k - \sum_{k=1}^K \hat{s}_k\right|\right] \leq q_1 \beta_1 + q_2 \beta_2,$$

where, β_1 and β_2 are given by

$$\beta_1 = 2 \sum_{\ell_1=1}^{M_1-1} \left(1 + \frac{1-2\ell_1}{M_1}\right) Q\left(\frac{(2\ell_1-1)}{2\sigma}\right), \quad (47a)$$

$$\beta_2 = 2 \sum_{\ell_2=1}^{M_2-1} \left(1 + \frac{1-2\ell_2}{M_2}\right) Q\left(\frac{(2\ell_2-1)|\rho|}{2\sigma}\right). \quad (47b)$$

Finally, recalling (43) and the fact that w_ψ is a strictly increasing function, we obtain

$$\mathbb{E}[|f - \hat{f}|] \leq w_\psi(q_1\beta_1 + q_2\beta_2).$$

Hence, we conclude the proof of Proposition 3.

D. Proof of Proposition 4

To analyze the complexity of the encoders \mathcal{E}_k , we note that \mathcal{E}_k and \mathcal{G}_ρ are the same for all the nodes, and they only involve four and two operations, respectively. Hence, their complexities are $\mathcal{O}(1)$, and they do not change the complexity order of the computations. Similarly, for corresponding decoder parts, i.e., \mathcal{D} , \mathcal{G}_ρ^{-1} , and have complexity of $\mathcal{O}(1)$. Furthermore, the complexity of the rounding operator \mathcal{Q} is also $\mathcal{O}(1)$ because it involves only basic arithmetic operations such as truncation and addition.

Therefore, we must compute the complexity only for the φ_k and ψ functions. To this end, we use the Taylor Series expansion to compute a general function with a complexity of $\mathcal{O}(n)$, where n represents the number of terms in the series. Because the required accuracy for given functions φ_k is at most $1/q$, we consider Taylor's remainder a computation error [50]. Hence, we can write the following for computing φ_k around point $s \in [-a, a]$:

$$\frac{|\varphi_k^{(m_k)}(s)|}{(m_k+1)!} (2a)^{(m_k+1)} \leq \frac{(2a)}{q}, \quad (48)$$

where $\varphi_k^{(m_k)}$ denotes the m_k -th derivation of function φ_k . Then, using the fact that $|\varphi_k^{(m_k)}(s)| \leq E_k$, we have

$$\frac{E_k}{(m_k+1)!} (2a)^{(m_k+1)} \leq \frac{2a}{q}, \quad (49)$$

$$\frac{E_k q}{2a} \leq \frac{(m_k+1)!}{(2a)^{(m_k+1)}}. \quad (50)$$

Then, by using Stirling approximation for $q \gg 1$ and doing manipulation [51], we reach

$$m_k \geq \left\lceil \frac{\ln\left(\frac{E_k q}{2a\sqrt{2\pi}}\right)}{\mathcal{W}\left((2ae)^{-1} \ln\left(\frac{E_k q}{2a\sqrt{2\pi}}\right)\right)} \right\rceil, \quad (51)$$

in which $\mathcal{W}(\cdot)$ is Lambert function defined as $\mathcal{W}(x) := z$ for $ze^z = x$ [52]. The number of operations must be calculated to compute the function φ_k . As a result, for computing all the encoders, we need to add them up, which leads to the number in (28). Following a similar procedure, we can show the complexity of computing the post-processing function ψ . Therefore, we conclude the proof.

REFERENCES

- [1] H. Tataria, M. Shafi, A. F. Molisch, M. Dohler, H. Sjöland, and F. Tufvesson, "6G wireless systems: Vision, requirements, challenges, insights, and opportunities," *Proc. IEEE*, vol. 109, no. 7, pp. 1166–1199, Jul. 2021.
- [2] H. Hellström et al., "Wireless for machine learning: A survey," *Found. Trends Signal Process.*, vol. 15, no. 4, pp. 290–399, 2022.
- [3] S. Daei, S. Razavikia, M. Kountouris, M. Skoglund, G. Fodor, and C. Fischione, "Blind asynchronous goal-oriented detection for massive connectivity," in *Proc. 21st Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, Aug. 2023, pp. 167–174.
- [4] M. Goldenbaum, H. Boche, and S. Stanczak, "Harnessing interference for analog function computation in wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 61, no. 20, pp. 4893–4906, Oct. 2013.
- [5] B. Nazer and M. Gastpar, "Computation over multiple-access channels," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3498–3516, Oct. 2007.
- [6] O. Abari, H. Rahul, and D. Katabi, "Over-the-air function computation in sensor networks," 2016, *arXiv:1612.02307*.
- [7] A. Şahin and R. Yang, "A survey on over-the-air computation," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 3, pp. 1877–1908, 3rd Quart., 2023.
- [8] S. Razavikia, J. M. B. da Silva, and C. Fischione, "Computing functions over-the-air using digital modulations," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2023, pp. 5780–5786.
- [9] M. Gastpar, "Uncoded transmission is exactly optimal for a simple Gaussian 'sensor' network," *IEEE Trans. Inf. Theory*, vol. 54, no. 11, pp. 5247–5251, Nov. 2008.
- [10] R. Soundararajan and S. Vishwanath, "Communicating linear functions of correlated Gaussian sources over a MAC," *IEEE Trans. Inf. Theory*, vol. 58, no. 3, pp. 1853–1860, Mar. 2012.
- [11] A. B. Wagner, S. Tavildar, and P. Viswanath, "Rate region of the quadratic Gaussian two-encoder source-coding problem," *IEEE Trans. Inf. Theory*, vol. 54, no. 5, pp. 1938–1961, May 2008.
- [12] L. Chen, N. Zhao, Y. Chen, F. R. Yu, and G. Wei, "Over-the-air computation for IoT networks: Computing multiple functions with antenna arrays," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 5296–5306, Dec. 2018.
- [13] F. Ang, L. Chen, N. Zhao, Y. Chen, and F. R. Yu, "Robust design for massive CSI acquisition in analog function computation networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2361–2373, Mar. 2019.
- [14] M. Goldenbaum and S. Stanczak, "Robust analog function computation via wireless multiple-access channels," *IEEE Trans. Commun.*, vol. 61, no. 9, pp. 3863–3877, Sep. 2013.
- [15] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 2022–2035, Mar. 2020.
- [16] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3546–3557, May 2020.
- [17] B. Nazer and M. Gastpar, "Compute-and-forward: Harnessing interference through structured codes," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6463–6486, Oct. 2011.
- [18] M. Goldenbaum, H. Boche, and S. Stanczak, "Nomographic functions: Efficient computation in clustered Gaussian sensor networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 4, pp. 2093–2105, Apr. 2015.
- [19] A. Sahin and R. Yang, "Over-the-air computation over balanced numerals," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2022, pp. 347–352.
- [20] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 491–506, Jan. 2020.
- [21] G. Zhu, Y. Du, D. Gündüz, and K. Huang, "One-bit over-the-air aggregation for communication-efficient federated edge learning: Design and convergence analysis," *IEEE Trans. Wireless Commun.*, vol. 20, no. 3, pp. 2120–2135, Mar. 2021.
- [22] A. Sahin, "Distributed learning over a wireless network with non-coherent majority vote computation," *IEEE Trans. Wireless Commun.*, vol. 22, no. 11, pp. 8020–8034, Nov. 2023.
- [23] L. You, X. Zhao, R. Cao, Y. Shao, and L. Fu, "Broadband digital over-the-air computation for wireless federated edge learning," *IEEE Trans. Mobile Comput.*, vol. 23, no. 5, pp. 5212–5228, May 2024.
- [24] A. Sahin, "Over-the-air majority vote computation with modulation on conjugate-reciprocal zeros," 2024, *arXiv:2405.02981*.

- [25] G. Mergen and L. Tong, "Type based estimation over multiaccess channels," *IEEE Trans. Signal Process.*, vol. 54, no. 2, pp. 613–626, Feb. 2006.
- [26] A. Pérez-Neira, M. Martínez-Gost, A. Şahin, S. Razavikia, C. Fischione, and K. Huang, "Waveforms for computing over the air," 2024, *arXiv:2405.17007*.
- [27] J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, "signSGD: Compressed optimisation for non-convex problems," in *Proc. ICML*, 2018, pp. 560–569.
- [28] M. Tang, S. Cai, and V. K. N. Lau, "Radix-partition-based over-the-air aggregation and low-complexity state estimation for IoT systems over wireless fading channels," *IEEE Trans. Signal Process.*, vol. 70, pp. 1464–1477, 2022.
- [29] L. Qiao, Z. Gao, M. B. Mashhadi, and D. Gündüz, "Massive digital over-the-air computation for communication-efficient federated edge learning," 2024, *arXiv:2405.15969*.
- [30] M. Goldenbaum, H. Boche, and S. Stańczak, "Reliable computation of nomographic functions over Gaussian multiple-access channels," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, May 2013, pp. 4814–4818.
- [31] J. Yao, W. Xu, Z. Yang, X. You, M. Bennis, and H. V. Poor, "Wireless federated learning over resource-constrained networks: Digital versus analog transmissions," 2024, *arXiv:2405.17759*.
- [32] S. Razavikia, J. M. B. da Silva, and C. Fischione, "ChannelComp: A general method for computation by communications," *IEEE Trans. Commun.*, vol. 72, no. 2, pp. 692–706, Feb. 2024.
- [33] J. Liu, Y. Gong, and K. Huang, "Digital over-the-air computation: Achieving high reliability via bit-slicing," 2024, *arXiv:2404.07121*.
- [34] X. Xie, C. Hua, J. Hong, and Y. Wei, "Joint design of coding and modulation for digital over-the-air computation," 2023, *arXiv:2311.06829*.
- [35] S. Razavikia, J. A. Peris, J. M. B. Da Silva, and C. Fischione, "Blind asynchronous over-the-air federated edge learning," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2022, pp. 1834–1839.
- [36] H. Hellström, S. Razavikia, V. Fodor, and C. Fischione, "Optimal receive filter design for misaligned over-the-air computation," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2023, pp. 1529–1535.
- [37] X. Cao, G. Zhu, J. Xu, and K. Huang, "Optimized power control for over-the-air computation in fading channels," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7498–7513, Nov. 2020.
- [38] D. Sprecher, "A representation theorem for continuous functions of several variables," *Proc. Amer. Math. Soc.*, vol. 16, no. 2, pp. 200–203, 1965.
- [39] R. C. Buck, "Approximate complexity and functional representation," *J. Math. Anal. Appl.*, vol. 70, no. 1, pp. 280–298, 1979.
- [40] D. A. Sprecher, "On the structure of continuous functions of several variables," *Trans. Amer. Math. Soc.*, vol. 115, p. 340, Mar. 1965.
- [41] K. Huber, "Codes over Gaussian integers," *IEEE Trans. Inf. Theory*, vol. 40, no. 1, pp. 207–216, Jan. 1994.
- [42] F. Hekland, P. A. Floor, and T. A. Ramstad, "Shannon-kotel-nikov mappings in joint source-channel coding," *IEEE Trans. Commun.*, vol. 57, no. 1, pp. 94–105, Jan. 2009.
- [43] Y. Shao, D. Gündüz, and S. C. Liew, "Federated edge learning with misaligned over-the-air computation," *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, pp. 3951–3964, Jun. 2022.
- [44] N. Aronszajn and P. Panitchpakdi, "Extension of uniformly continuous transformations and hyperconvex metric spaces," *Pacific J. Math.*, vol. 6, no. 3, pp. 405–439, Sep. 1956.
- [45] Y. Ye, *Interior Point Algorithms: Theory and Analysis*. Hoboken, NJ, USA: Wiley, 2011.
- [46] Z.-Q. Luo, W.-k. Ma, A. M. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.
- [47] D. E. Knuth, "Sorting and searching," in *The Art of Computer Programming*, vol. 3. Reading, MA, USA: Addison-Wesley, 1998.
- [48] A. Goldsmith, *Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
- [49] E. J. McShane, "Jensen's inequality," *Bull. Amer. Math. Soc.*, vol. 43, pp. 521–527, 1937.
- [50] G. B. Folland, "Remainder estimates in Taylor's theorem," *The Amer. Math. Monthly*, vol. 97, no. 3, pp. 233–235, 1990.
- [51] F. K. Ameyou, "Properties and computation of the inverse of the Gamma function," M.S. thesis, Dept. Math., Univ. Western Ontario, London, ON, Canada, 2018.
- [52] R. M. Corless, G. H. Gonnet, D. E. Hare, D. J. Jeffrey, and D. E. Knuth, "On the Lambert W function," *Adv. Comput. Math.*, vol. 5, pp. 329–359, Dec. 1996.



Saeed Razavikia (Member, IEEE) is currently pursuing the Ph.D. degree with the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, funded by the Wallenberg AI, Autonomous Systems and Software Program (WASP). His current research interests include optimization, the theory of machine learning, and statistical signal processing.



José Mairton Barros da Silva Jr. (Member, IEEE) received the B.Sc. (Hons.) and M.Sc. degrees in telecommunications engineering from the Federal University of Ceará, Brazil, in 2012 and 2014, respectively, and the Ph.D. degree from the KTH Royal Institute of Technology, Stockholm, Sweden, in 2019. From 2019 to 2021, he was a Post-Doctoral Researcher at the KTH Royal Institute of Technology. From 2022 to 2023, he was a Marie Skłodowska-Curie Post-Doctoral Fellow with Princeton University, Princeton, NJ, USA, and the KTH Royal Institute of Technology. He is currently an Assistant Professor with the Division of Computer Systems, Uppsala University, Sweden. His research interests include distributed machine learning and optimization over wireless communications. He is also a Workshops, a Tutorials, and a Symposia Officer of the IEEE Communications Society Emerging Technology Initiative on Machine Learning for Communication. He has been involved in the organization of many conferences and workshops, including co-chairing IEEE ICMLCN 2024 and IEEE SECON 2022–2023. He gave several tutorials at many IEEE flagship conferences, including ICASSP, PIMRC, ICC, and GLOBECOM.



Carlo Fischione (Senior Member, IEEE) received the Laurea degree (summa cum laude) in electronic engineering and the Ph.D. degree in electrical and information engineering from the University of L'Aquila, Italy, in April 2001 and May 2005, respectively. He has held various research positions, such as a Visiting Professor with Massachusetts Institute of Technology, Cambridge, MA, USA, in 2015; an Associate Professor with Harvard University, Cambridge, in 2015; and a Visiting Scholar with the University of California at Berkeley, CA, USA, from 2004 to 2005, where he was a Research Associate from 2007 to 2008. He is currently an Honorary Professor with the Department of Mathematics, Information Engineering, and Computer Science, University of L'Aquila. He is the Co-Founder and a Scientific Advisor of ELK.Audio. He is also a Full Professor with the Department of Electrical Engineering and Computer Science, Division of Network and Systems Engineering, KTH Royal Institute of Technology, Stockholm, Sweden. He is the Director of the KTH-Ericsson Data Science Micro Degree Program directed to Ericsson globally, the Director of the undergraduate education at NSE. He received the Starting Grant of the Swedish Research Council in 2008. His research interests include applied optimization, wireless Internet of Things, and machine learning. He received a number of awards, such as the "IEEE Communication Society S. O. Rice" Award for the Best IEEE TRANSACTIONS ON COMMUNICATIONS Paper of 2018 and the Best Paper Award of IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS in 2007. He is the Chair of the IEEE Machine Learning for Communications Emerging Technologies Initiative, and the founding General Chair of the IEEE International Conference on Machine Learning for Communications and Networking—IEEE ICMLCN 2024. He is an Editor of IEEE TRANSACTIONS ON COMMUNICATIONS (Machine Learning for Communications Area) and IEEE TRANSACTIONS ON MACHINE LEARNING FOR COMMUNICATION AND NETWORKING and has served as an Associate Editor of *Automatica* (IFAC) from 2014 to 2019. He is an Ordinary Member of DASP (the Italian academy of history Deputazione Abruzzese di Storia Patria). He is a Distinguished Lecturer of the IEEE Communication Society.