A Survey on Over-the-Air Computation

Alphan Şahin, Member, IEEE, and Rui Yang, Member, IEEE

Abstract—Communication and computation are often viewed as separate tasks. This approach is very effective from the perspective of engineering as isolated optimizations can be performed. However, for many computation-oriented applications, the main interest is a function of the local information at the devices, rather than the local information itself. In such scenarios, information theoretical results show that harnessing the interference in a multiple access channel for computation, i.e., over-the-air computation (OAC), can provide a significantly higher achievable computation rate than separating communication and computation tasks. Moreover, the gap between OAC and separation in terms of computation rate increases with more participating nodes. Given this motivation, in this study, we provide a comprehensive survey on practical OAC methods. After outlining fundamentals related to OAC, we discuss the available OAC schemes with their pros and cons. We provide an overview of the enabling mechanisms for achieving reliable computation in the wireless channel. Finally, we summarize the potential applications of OAC and point out some future directions.

Index Terms—Over-the-air computation.

I. INTRODUCTION

O VER-THE-AIR computation (OAC) refers to the computation of mathematical functions by exploiting the signal superposition property of wireless multiple access channels. The distinct feature of OAC is that the local data at the edge devices (EDs), such as smartphones, laptops, tablets, vehicles, or sensors, are not acquired over orthogonal channels to perform a computation task at a fusion node, e.g., an edge server (ES) at a base station or an access point. Instead, the computation is handled by harnessing the interference via simultaneous transmissions. For example, suppose that the goal is to evaluate a function \( f(s_1, \ldots, s_K) \) at an ES, where \( s_k \) is the symbol at the \( k \)-th ED. With the separation of communication and computation tasks, the function is computed at the fusion node after each symbol is received via orthogonal or non-orthogonal resources (i.e., orthogonal multiple access (OMA) and non-orthogonal multiple access (NOMA)), as illustrated for time-domain multiple access (TDMA) in Fig. 1(a). On the other hand, with OAC, the function is intended to be computed through signal superposition in the channel as shown in Fig. 1(b). In this example, the key observation is that if the ES is not interested in the local information but only in a function of them, OAC paves the way for reducing resource usage, which otherwise scales with the number of EDs. Hence, it is a fundamental and disruptive concept to the traditional way of handling computation and communication tasks independently.

The idea of function computation over a multiple access channel was first thoroughly analyzed in Bobak’s pioneering work in [1] and the theoretical limits of computation over multiple access channels were investigated for a fixed many-to-one function. In [2], Goldenbaum made the first connection between nomographic functions and OAC. In [3] and [4], it was shown that OAC can provide a significantly higher achievable computation rate than separating communication and computation. Given the promising information theoretical results, OAC has been drawing more and more attention in the literature. Initially, it has been applied to communication problems in the interference channel, e.g., physical layer network coding [5], [6], compute-and-forward relaying strategy [7], and wireless sensor networks (WSNs) to address the issues like acceleration in gossip networks [8] and several computation tasks [9], [10]. With the increased interest in applications that require heavy computation, it has recently been utilized in multi-disciplinary fields such as machine learning over wireless networks [11], wireless control systems [12], and computing frameworks like wireless data centers [13] and wireless intra-chip computations [14].

The exciting applications have led to the investigation of OAC from various perspectives and resulted in a wide-variety of computation strategies. This paper aims to discuss
these OAC schemes without losing the mathematical rigor and how these methods handle the challenges such as the detrimental impact of wireless channels on computation, synchronization errors, maintaining accurate and fresh channel state information (CSI) at the radios, security, and hardware impairments such as power amplifier non-linearity.

A. Relation to Other Surveys and Our Contributions

The reader can find relevant discussions on distributed inference over sensor networks in [15]. The methods relying on compute-and-forward relaying scheme and uncoded strategies for physical layer network coding are comprehensively discussed and compared in [16], [17]. To reduce the per-round communication latency for the implementation of distributed learning over a wireless network, OAC has been used in many recent works as an enabler. We refer the readers interested in wireless systems for machine learning in general to the excellent survey papers in [18], [19], [20], [21], [22], [23], [24], [25] and the references therein. In [24], federated edge learning (FEEL), i.e., implementation of federated learning (FL) [26] over a wireless network, and the resource management for FEEL are surveyed. In [11], several exciting applications of OAC and research directions in this area are discussed without mathematical details. In [27], semantic communication is thoroughly surveyed and OAC is mentioned as one of the task-oriented semantic communication paradigms. In [28], [29], OAC is particularly analyzed from the perspective of integrated sensing, communication, and computation. In [30], over-the-air distributed computing for artificial intelligence applications is envisioned for 6G wireless networks. In [31], the particular interest is in the applications that enjoy signal superposition in general. Besides OAC, the topics such as NOMA, interference alignment, multiple antenna systems, signal superposition in general. Besides OAC, the topics such as edge computing and off-loading are surveyed. We also acknowledge [33] which discusses the OAC from the perspective of various network architectures and provides an excellent survey on the OAC based on multiple antennas at the devices.\(^1\)

The main focus of this study is to investigate \textit{how to compute a function over a wireless network reliably and efficiently}. Our priority is to form a composition that can provide a relative comparison of the state-of-the-art OAC techniques with pros and cons, particularly from the perspective of the physical layer of communication systems. Since a wide variety of applications can benefit from the OAC, in this study, we focus on the computation itself, rather than a particular application. We seek answers to three main questions:

1) \textbf{What functions can potentially be calculated with OAC?}

To answer this question, we review the nomographic functions that appear in both mathematics and communication literature.

2) \textbf{What are the OAC schemes in the state-of-the-art and their trade-offs to deal with the distortion in wireless channels?}

To address this question, we first give a general system model along with fundamental metrics on OAC. Under this framework, we evaluate the methods based on how they achieve computation under the fading channel and the encoding strategies.

3) \textbf{What are the mechanisms that play a role in achieving a reliable OAC?}

To answer this question, we review the impacts of synchronization impairments, power management, and channel estimation on OAC and elaborate on security aspects and computation architectures for OAC. Finally, we provide an overview of the applications of OAC in the literature and point out the potential areas that can be improved for OAC.

\textbf{Organization:} The rest of the study is organized as follows. In Section II, we provide an overview of the fundamentals and discuss the functions that can potentially be computed via OAC. In Section III, we discuss the state-of-the-art OAC schemes, comprehensively. In Section IV, we discuss the enabling mechanisms to achieve a reliable computation. We summarize the potential applications of OAC in various fields in Section V. We finalize our discussions with various topics that need to be investigated further in Section VI.

\textbf{Notation:} The complex and real numbers are denoted by \(\mathbb{C}\) and \(\mathbb{R}\), respectively. The \(K\)-times Cartesian product of space \(A\) is shown as \(A^K\). \(F(A)\) represents the space of every function that maps \(A\) to \(\mathbb{R}\). \(E\{\cdot\}\) denotes the expectation over all random variables. The function \(\text{sign}(\cdot)\) results in 1, \(-1\), or 0 for a positive, a negative, or a zero-valued argument, respectively. The symbol \(\otimes\) denotes linear convolution. The function \(\mathbb{I}_{\{\cdot\}}\) results in 1 if its argument holds, otherwise it is 0. \(\text{Pr}(\cdot)\) is the probability of an event. The zero-mean multivariate complex Gaussian distribution with the covariance matrix \(C_M\) of an \(M\)-dimensional random column vector \(x \in \mathbb{C}^M\) is denoted by \(x \sim \mathcal{CN}(0_M, C_M)\). \(\mathcal{N}(\mu, \sigma^2)\) is the normal distribution with the mean \(\mu\) and the variance \(\sigma^2\). The trace of a matrix is denoted by \(\text{tr}\{\cdot\}\). The continuous uniform distribution is denoted by \(U_{[a, b]}\), where \(a\) and \(b\) are the minimum and the maximum values, respectively. The function \(\log_2^+ (x)\) is defined as \(\max(\log_2(x), 0)\). Kronecker delta is expressed as \(\delta_{ij}\).

II. \textbf{What Can Be Calculated With OAC?}

OAC aims to compute a multivariate function by relying on its representation that can structurally match with the underlying operation that multiple access channel naturally performs. In wireless communications, multiple access channels are modeled with additive property, i.e., the signal superposition. With this property, the OAC problem boils down to the representation of a target function with a special function, called a \textit{nomographic} function, or a set of nomographic functions over multiple wireless resources. These functions are called nomographic because they are inline with the nomographs that solve certain equations through some graphs, i.e., analog computing. A well-known example of a nomograph is the Smith chart which assists in solving problems related

\(^1\)Our paper and [33] are independently developed and compensate each other from the perspective of classifications of available OAC approaches. The corresponding pre-prints were listed on arXiv.org one day apart (October 19, 2022).
to transmission lines. While the nomographs allow quick and accurate computations, the use cases of nomographs diminished historically due to the effectiveness of digital computers. Nevertheless, the fundamental theories about nomography are intricate, arguably connected to the neural networks, and paved the way for addressing the scenarios where digital computation suffers from latency, power consumption, and limited-communication bandwidth. In this section, we discuss the preliminaries on nomographic functions to reveal what can be calculated with OAC.

A. Preliminaries

Definition 1 (Nomographic Function [2], [34], [35], [36]): Let $S^K$, $K \geq 2$, be a compact metric space. A function $f : S^K \to \mathbb{R}$ for which there exist functions $\psi_k \in F(S)$, $k \in \{1, \ldots, K\}$, and $\varphi \in F(\mathbb{R})$ such that $f$ can be represented as

$$f(s_1, s_2, \ldots, s_K) = \varphi \left( \sum_{k=1}^{K} \psi_k(s_k) \right),$$

(1)

is called nomographic function and $\mathcal{N}(S^K)$ is the space of nomographic functions with the domain $S^K$.

The functions $\psi_k$, $\forall k$, and the function $\varphi$ are further called pre-processing functions (or inner functions) and post-processing function (or outer function), respectively. Equation (1) reveals why a nomographic function is relevant to OAC: Equation (1) can be interpreted as an evaluation of the function $f$ in an ideal uplink (UL) channel (i.e., no noise, no multi-path channel distortion), where $s_k$ and $\psi_k$ are the symbols and the pre-processing functions at $k$th data-generating node, respectively, the sum of the signals from $K$ nodes corresponds to the superposition that naturally occurs in the channel, and $\varphi$ is the post-processing function at the fusion center. To the best of our knowledge, this connection is first made in Goldenbaum’s work in [2], [34], [35], [36] while the non-linear function examples in the form of (1) appear in [37], [38], [39] without discussing the family of nomographic functions.

It is worth noting that the compactness mentioned in Definition 1 is an important assumption, especially in the analysis of continuous functions. For example, the range of a continuous function $f(s_1, s_2, \ldots, s_K)$ on a compact space $S^K$ is compact. Since the function is bounded, one can ensure that the limits exist, or that suprema and infima are taken by the function. If the space is not compact, it can be harder to analyze the behavior of a given function and more structural properties related to the function need to be known. From the perspective of OAC, compactness is inherited due to practical limitations. For instance, the measure space of a sensor is typically compact because a sensor can quantify values in a finite closed interval, e.g., $0^\circ C \leq s_k \leq 100^\circ C$, $\forall k$. Hence, to make general statements about entire function spaces and not only about specific examples, the space $S^K$ in Definition 1 is considered to be compact.

Now, let us denote the space of nomographic functions, the space of nomographic functions with the restriction of continuous pre- and post-processing functions, and the space of continuous functions with the domain $\mathbb{R}^K$ as $\mathcal{N}(\mathbb{R}^K)$, $\mathcal{N}^0(\mathbb{R}^K)$, and $\mathcal{C}^0(\mathbb{R}^K)$, respectively. Sprecher and Buck provide insights into the representation of a function $f \in \mathcal{C}^0(\mathbb{R}^K)$ as a nomographic function as follows:

Theorem 1 (Sprecher’65 [40]): Every function $f \in \mathcal{C}^0(\mathbb{R}^K)$ can be represented with real, monotonic increasing pre-processing functions and possibly a discontinuous post-processing function.

Theorem 2 (Buck’79 [41]): Every function $f \in \mathcal{F}(\mathbb{R}^K)$ is nomographic (i.e., $\mathcal{N}(\mathbb{R}^K) = \mathcal{F}(\mathbb{R}^K)$).

The key idea for the proof of Theorem 2 is to show there exists a one-to-one mapping from $\mathbb{R}^K$ to a space $\Gamma \subset \mathbb{R}$ in the form of $g(s_1, \ldots, s_K) = \sum_{k=1}^{K} \psi_k(s_k)$. Given the existence of such $g$ (therefore, the pre-processing functions exist), the post-processing function can then be expressed as $\varphi(x) = f(g^{-1}(x))$, where $g^{-1}$ is the inverse function that maps $x \in \Gamma$ to $(s_1, \ldots, s_K)$. Without any restriction on the pre-functions and the post-processing function, such a map can be obtained by choosing $\Gamma = \mathbb{E}$ and constructing the binary representation of $x \in \Gamma$ by uniformly interleaving the digits of the binary representations of the symbol $s_k$, $\forall k$ (see [41, p. 287] and [42, p. 2]). For this specific constructive proof, $\psi_k$ relies on reading the binary representation of $s_k$ in base $2^K$, which implicitly causes discontinuity in its range. The proof also shows the existence of special nomographic functions with an interesting property.

Definition 2 (Universality): The pre-processing functions are universal if they are fixed and can be used to calculate every function in $\mathcal{F}(\mathbb{E}^K)$.

The universality is a desirable property for OAC because the pre-processing functions do not need to be re-designed (i.e., less communication overhead) if the target function changes over time. This property is exploited in [2], [34] for multi-cluster computation as discussed in Section IV-C. It is also mentioned that universality provides robustness against changes in network topology (via dropping and joining devices) in the sense that transmitting nodes do not need to adapt their pre-processing functions.

If one desires the pre- and post-processing functions to be continuous for an arbitrary continuous function $f$, Theorem 2 is unfortunately not valid:

Theorem 3 (Buck’82 [43]): $\mathcal{N}^0(\mathbb{E}^K)$ is nowhere dense in $\mathcal{C}^0(\mathbb{E}^K)$.

A canonical example of Theorem 3 is geometric mean, i.e., $f(s_1, s_2, \ldots, s_K) = (\prod_{k} s_k)^{\frac{1}{K}}$. This function cannot be represented as $\varphi(\sum_{k=1}^{K} \psi_k(s_k))$ with the continuous functions $\psi_1, \ldots, \psi_K, \varphi$ on $\mathbb{E}$ as demonstrated for $K = 2$ by Arnold’44 and for an arbitrary $K$ by Goldenbaum et al. [2]. Theorem 3 implies that there exist infinite number of continuous functions in $\mathcal{C}^0(\mathbb{E}^K)$ that cannot be approximated with a nomographic function in $\mathcal{N}^0(\mathbb{E}^K)$ for a given arbitrary precision. Kolmogorov remarkably addresses the issue of representing a continuous function with a set of nomographic functions in $\mathcal{N}^0(\mathbb{E}^K)$:

Theorem 4 (Kolmogorov’57 [45]): Every function $f \in \mathcal{C}^0(\mathbb{E}^K)$ can be represented as the superposition of at most

2 Nomographic functions in mathematics are often investigated by defining the compact space $S$ as $\mathbb{E}$. 

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
$2K + 1$ nomographic functions in $N^0(E^K)$, i.e.,

$$f(s_1, s_2, \ldots, s_K) = \sum_{\ell=1}^{2K+1} \varphi_{\ell} \left( \sum_{k=1}^{K} \psi_{\ell k}(s_k) \right), \quad (2)$$

where the post-processing functions $\varphi_{\ell}$ depend on $f$ and the functions $\psi_{\ell k}$ are independent of $f$.

Geometrically, the $2K + 1$ inner sums in (2) ensure the existence of a continuous and bijective correspondence between $(s_1, \ldots, s_K) \in E^K$ and $(\varphi_1(\sum_{k=1}^{K} \psi_{1 k}(s_k)), \ldots, \varphi_{2K+1}(\sum_{k=1}^{K} \psi_{2K+1 k}(s_k))) \in \mathbb{R}^{2K+1}$. Hence, the inner sums describe a homeomorphism that continuously embeds $E^K$ into $\mathbb{R}^{2K+1}$. In [46], Sternfeld enhances the statement of Theorem 4 by showing that the $2K + 1$ nomographic functions in (2) cannot be reduced to represent every $f \in C^0(E^K)$. Hence, from the perspective of OAC, Theorem 4 implies that at least $2K + 1$ wireless resources need to be allocated where each resource is dedicated to a nomographic function in $N^0(E^K)$ to calculate every function in $C^0(E^K)$.

In mathematics, Theorem 4, also known as Kolmogorov’s superposition or Kolmogorov-Arnold representation theorem, is notable because it solves a more constrained (i.e., the function $f$ needs to be continuous), but a more general form (i.e., the superposition of only one variable functions) of Hilbert’s the thirteenth problem in [47]. There are also other variants of Kolmogorov’s superposition and constructive proofs that show how to obtain the pre- and post-processing functions. For a comprehensive discussion on the variants and constructions, we refer the reader to [48, Ch. 2]. A variant that is mentioned in the OAC literature [2] as follows.

Theorem 5 (Braun’09 [49]): For every function $f \in C^0(E^K)$, there exist $2K + 1$ nomographic functions in $N^0(E^K)$ such that

$$f(s_1, s_2, \ldots, s_K) = \sum_{\ell=1}^{2K+1} \varphi_{\ell} \left( \sum_{k=1}^{K} \alpha_k \psi_{\ell k}(s_k) + (\ell - 1)\beta \right), \quad (3)$$

where the pre-processing function $\psi$ is a well-defined, continuous, monotone, and independent of $f$, the coefficients $\alpha_k$, $\forall k$, and $\beta$ are appropriate non-negative real constants.

The key observation made in [2] based on Theorem 5 is that to calculate every function in $C^0(E^K)$ with continuous nomographic functions over $2K + 1$ resources, the pre-processing functions can be designed to be universal. Note that the superposition in (3) involves $2K + 1$ post-processing function and one single pre-processing function in the literature, it is shown that the superposition can also be expressed with a single pre-processing function and a single post-processing function as discussed in [48, Th. 1] and [50, Th. 2.14] by introducing a shift to the arguments of the post-processing functions in (3). Also, Kolmogorov’s superposition can be interpreted as a special feed-forward neural network and is useful to predict the complexity of neural networks (see the discussions in [51], [52], [53]).

In some cases, it may be desirable not to consume $2K + 1$ wireless resources to calculate a specific continuous function with $2K + 1$ continuous nomographic functions. In this case, one may follow one of two different directions: Manipulating the domain of the target function or constructing a nomographic function that approximates the target function. In the first approach, some part of the domain is cut out so that the nomographic function can be calculated with continuous pre- and post-processing functions. For instance, if $S$ is chosen as $[\alpha, 1]$ for $\alpha \in (0, 1)$, the geometric mean can be calculated with a nomographic function with $\psi_k(x) = \ln(x)$, $\forall k$, and $\varphi(x) = e^{x/\alpha}$ on $S$. In the second approach, a nomographic approximation can be defined as follows [2]:

**Definition 3 (Nomographic Approximation):** Let $\epsilon > 0$ be an arbitrary constant. The space of approximable nomographic functions with respect to the precision $\epsilon$ is defined by

$$N^0_\epsilon(E^K) \triangleq \left\{ f \in F(E^K) \mid \exists (\psi_1, \ldots, \psi_K, \varphi) \in C^0(E) \times \cdots \times C^0(E) \times C^0(\mathbb{R}) : \left\| f - \varphi \left( \sum_{k=1}^{K} \psi_k(s_k) \right) \right\|_\infty \leq \epsilon \right\}. \quad (4)$$

If $f \in N^0_\epsilon(E^K)$, we write $f(s_1, \ldots, s_K) \approx \varphi(\sum_{k=1}^{K} \psi_k(s_k))$. For example, under Definition 3, the geometric mean on $E^K$ is a function in $N^0_\epsilon(E^K)$ because it can be approximated with $\psi_k(x) = \ln(x + 1/p_0(\epsilon))$ and $\varphi(x) = e^{x/\alpha}$ for $p_0(\epsilon) > 0$. Nevertheless, a complete characterization of the approximate nomographic functions is still an area that requires more investigation as it is possible to define the space of approximable nomographic functions in different ways. For instance, in [54, eq. (5)], an approximate nomographic function is defined in a stochastic manner. For further theoretical investigations on approximate nomography, the reader is also referred to [54], [55], [56], [57].

Another interesting function space is the class of symmetric functions elaborated in [58].

**Definition 4 (Symmetric Function):** Let $\sigma : S^K \to S^K$ denotes a permutation. A function $f : S^K \to \mathbb{R}$ that is invariant with respect to permutations of its arguments, i.e.,

$$f(s_1, \ldots, s_K) = f(\sigma(s_1, \ldots, s_K)), \quad \forall \sigma \quad (5)$$

is called a symmetric function.

A distinct feature of the space of symmetric function is that only the data itself is important, rather than its origin. From an application standpoint, many functions such as mean, maximum, minimum, median, and majority vote (MV) that have either exact or approximate nomographic functions belong to this class. The second important feature is that the functions in this space can be calculated through the type function, i.e., frequency histogram [9], [10], [58], [59]. Type function can be defined as multiple weighted arithmetic sums of indicator functions, i.e., counting the number of devices based on a certain set, which is also investigated under type-based multiple access (TBMA) in [9], [10].

**B. Common Nomographic Functions**

In Table I, we list several exact and approximate nomographic functions discussed in the literature. While arithmetic mean, weighted sum, and MV are used in distributed learning applications, modulo-2 sum often appears in physical
layer networking coding. The product operation is used for key generation in [60]. The maximum, minimum, and counting functions are used in WSN (e.g., generating an alert if the temperature rises) or to calculate histogram (e.g., calculating measurement statistics with TBMA [9], [10]). Geometric mean, p-norm, and polynomial functions are often mentioned to provide nomographic function examples that can be calculated over a wireless network. In particular, p-norm is used for computing average-pooling or max-pooling over the air in [61].

An interesting direction is to calculate an approximate nomographic function with a continuous and monotone post-processing function and continuous pre-processing functions for a given continuous function. In [56], an approximation is obtained by using a combination of a dimensionwise function decomposition. In this approach, the target function is skewed with a bijective function such that the resulting function can be approximated well with a first-order analysis of variance (ANOVA) decomposition. To calculate the skew function, Bernstein polynomials are used. It is worth noting that Bernstein polynomials can be utilized to constructively prove the Weierstrass approximation theorem that states every continuous function can be approximated with an arbitrary precision over any finite interval by a polynomial of a sufficient order.

Another interesting direction is to calculate the target function by expressing it as a solution to an optimization problem and solving the problem through iterations that can be expressed with some elementary nomographic functions. For example, as discussed in Section IV-E1, the geometric median can be calculated through iterations over-the-air by using the Weiszfeld algorithm in [62]. In [63], [64], and [65], by using the binary representations of the parameters, several non-linear functions, e.g., maximum or minimum, are proposed to be calculated through the communication rounds. For instance, to calculate the maximum of the parameters, in the first round, the ES inquires to the EDs with bit 1 in the most significant bit position of the binary representation of the parameter. If there is any response to the inquiry, the ES detects that the most significant bit of the maximum of the parameters is 1; otherwise, it is 0. In the second round, if the most significant bit is detected as 1, the ES inquires to the EDs with bit 1 in both the most and the second significant bits. Otherwise, the ES inquires to all EDs with bit 1 in the second significant bit position. From the responses to the second inquiry, the ES determines the second significant bit. The procedure continues until the least significant bit is detected. The same procedure can be used for computing minimum function by using the reciprocal of the parameters. The reader is also referred to [59] for successive partitions to compute functions.

### III. WHAT ARE THE OAC SCHEMES?

An OAC scheme aims to realize (1) (or (2)) over a wireless multiple-access channel (MAC) with a fidelity criterion. For a rigorous classification and a generalization of the OAC schemes in the state-of-the-art, consider an OAC scheme that targets to calculate the nomographic function $z[n] = f(s[n])$ for the symbol vector $s[n] = [s_1[n], \ldots, s_K[n]]^T$ for $(K)$, can be approximated well with a first-order analysis of variance (ANOVA) decomposition. To calculate the skew function, Bernstein polynomials are used. It is worth noting that Bernstein polynomials can be utilized to constructively prove the Weierstrass approximation theorem that states every continuous function can be approximated with an arbitrary precision over any finite interval by a polynomial of a sufficient order.

Another interesting direction is to calculate the target function by expressing it as a solution to an optimization problem and solving the problem through iterations that can be expressed with some elementary nomographic functions. For example, as discussed in Section IV-E1, the geometric median can be calculated through iterations over-the-air by using the Weiszfeld algorithm in [62]. In [63], [64], and [65], by using the binary representations of the parameters, several non-linear functions, e.g., maximum or minimum, are proposed to be calculated through the communication rounds. For instance, to calculate the maximum of the parameters, in the first round, the ES inquires to the EDs with bit 1 in the most significant bit position of the binary representation of the parameter. If there is any response to the inquiry, the ES detects that the most significant bit of the maximum of the parameters is 1; otherwise, it is 0. In the second round, if the most significant bit is detected as 1, the ES inquires to the EDs with bit 1 in both the most and the second significant bits. Otherwise, the ES inquires to all EDs with bit 1 in the second significant bit position. From the responses to the second inquiry, the ES determines the second significant bit. The procedure continues until the least significant bit is detected. The same procedure can be used for computing minimum function by using the reciprocal of the parameters. The reader is also referred to [59] for successive partitions to compute functions.

### III. WHAT ARE THE OAC SCHEMES?

An OAC scheme aims to realize (1) (or (2)) over a wireless multiple-access channel (MAC) with a fidelity criterion. For a rigorous classification and a generalization of the OAC schemes in the state-of-the-art, consider an OAC scheme that targets to calculate the nomographic function $z[n] = f(s[n])$ for the symbol vector $s[n] = [s_1[n], \ldots, s_K[n]]^T$, $\forall n \in \{1, \ldots, N_I\}$, $N_I \geq 1$. Let $s_k = [s_k[1], \ldots, s_k[N_I]]^T$ and $p_k = [p_k[1], \ldots, p_k[N_I]]^T$ denote the symbol vector and the pre-processed symbol vector at the $k$th ED for $p_k[n] = \varphi(s_k[n])$ for $\forall n \in \{1, \ldots, N_I\}$. The $k$th ED calculates the encoded vector $c_k \in \mathbb{C}^B$ as

$$c_k = [c_1, \ldots, c_B]^T = \epsilon_k(p_k),$$

where $\epsilon_k : \mathbb{R}^{N_I} \rightarrow \mathbb{C}^B$ is the encoder (e.g., source encoder, channel encoder, constellation mapping, or a combination of these operations) and $B$ is the number of modulation symbols in a complex-valued codeword.

Let $\mathcal{M}$ be a resource mapper that maps $B$ modulation symbols to the $B$ available resources. Now, consider $L$ modulation symbols, denoted by $m_k = [m_k[1], \ldots, m_k[L]]^T$, that are processed with a linear precoder $B_k \in \mathbb{C}^{N_I \times L}$ as

$$x_k = [x_k[1], \ldots, x_k[N_I]]^T = B_k m_k,$$

where $x_k \in \mathbb{C}^{N_I}$ is the transmitted symbols from the $k$th ED over $N_I$ dimensions. Hence, each ED applies $N_{\text{access}} \triangleq B/L$ linear precoders to the modulation symbols in total.

The received vector at the ES, denoted by $y \in \mathbb{C}^{N_I}$, can be written as

$$y = [y_1, \ldots, y_{N_I}]^T = \sum_{k=1}^K \sqrt{P_k} H_k x_k + n,$$

where $H_k \in \mathbb{C}^{N_I \times N_I}$ is the channel matrix between the $k$th ED and the ES with the assumptions of $E\{H_k H_k^H\} = N_0 I_{N_I}$ and each element of $H_k$ is modeled as a zero-mean symmetric complex Gaussian random variable, unless otherwise
Fig. 2. A generalized model for an OAC method. The domains of the transmitted vectors $x_k$, $\forall k$, and the received vector $y$ can be time, frequency, space, etc. depending on the OAC method.

stated, $n = [n_1, \ldots, n_{N_t}]^T \in \mathbb{C}^{N_t}$ is the zero-mean symmetric noise vector with the variance $\sigma_n^2$, $N_t$ is the number of available dimensions at the ES, and $P_k \in \mathbb{R}$ denotes the received signal power for the $k$th ED, which is a function of the large-scale channel model, power control, waveform, power amplifier (PA) non-linearity, and adjacent-channel-leakage ratio (ACLR) requirements (see Section IV-B for further discussions). The receiver at the ES processes the superposed vector $y$ with a linear decoder $A^H \in \mathbb{C}^{L \times N_f}$ (e.g., to overcome the impact of the channel on the superposition along with $B_k$, $\forall k$) as

\[
\hat{m} = [\hat{m}_1, \ldots, \hat{m}_L]^T = A^H y
\]

By using $N_{\text{access}}$ outputs of the linear decoders, the resource de-mapper $M^{-1}$ first constructs the superposed codeword $\hat{c} = [\hat{c}_1, \ldots, \hat{c}_B]^T \in \mathbb{C}^B$. Afterward, the receiver calculates an estimate of the superposed pre-processed symbols as

\[
\hat{p} = [\hat{p}[1], \ldots, \hat{p}[N_f]]^T = \delta(\hat{c}),
\]

where $\delta : \mathbb{C}^B \rightarrow \mathbb{R}^{N_f}$ is the decoder (e.g., constellation de-mapper, channel decoder, source decoder, or a combination of these operations) at the ES. Finally, the target functions are evaluated with the post-processing function as $z = [z[1], \ldots, z[N_f]]^T \in \mathbb{R}^{N_f}$ for $z[n] = \varphi(\hat{p}[n]) \in \mathbb{R}$, $\forall n \in \{1, \ldots, N_f\}$.

In Fig. 2, we provide a block diagram based on aforementioned transmitter and receiver operations for OAC. It is worth noting that we do not specify the domains of the transmitted vectors $x_k$, $\forall k$, and the received vector $y$. Without loss of generality, the domain can be time, frequency, or space, depending on the scheme. In addition, if the target function is based on Kolmogorov’s superposition, $2K + 1$ nomographic functions in (2) can be computed over orthogonal resources (see [36] for an example) with a general OAC scheme.

A. Metrics

In this subsection, we discuss widely used metrics for assessing OAC schemes.

1) Error Definitions: Let $\hat{f}(s[n])$ be an estimate of $f(s[n])$ for $s[n] = [s_1[n], \ldots, s_K[n]]^T$, $n \in \{1, 2, \ldots, N_f\}$. The function-estimation error (FEE) can be expressed as

\[
\epsilon_{\text{FEE}}(\hat{f}(s[n])) = |\hat{f}(s[n]) - f(s[n])|.
\]

In [39] and [66], $f(s[n]) \in [f_{\text{min}}, f_{\text{max}}]$, $\forall n$, a normalized FEE is defined by

\[
\epsilon_{\text{FEE}}(\hat{f}(s[n])) = \frac{\epsilon_{\text{FEE}}(\hat{f}(s[n]))}{f_{\text{max}} - f_{\text{min}}}.
\]

2) Average Error: The classical mean-squared error (MSE) [67], [68], normalized MSE, and Bayesian MSE for computation can be expressed as

\[
\text{MSE}(\hat{f}(s[n])) = \mathbb{E}\left\{||\epsilon_{\text{FEE}}(\hat{f}(s[n]))||_2^2\right\},
\]

and

\[
\text{NMSE}(\hat{f}(s[n])) = \frac{\mathbb{E}\left\{||\epsilon_{\text{FEE}}(\hat{f}(s[n]))||_2^2\right\}}{||f(s[n])||_2^2},
\]

and

\[
\text{BMSE} = \mathbb{E}\left\{\text{MSE}(\hat{f}(s[n]))\right\},
\]

respectively. In [64], the mean-squared function error (MSFE) is defined by

\[
\text{MSFE} = \mathbb{E}\left\{||\epsilon_{\text{FEE}}(\hat{f}(s[n]))||_2^2\right\},
\]

where the expectation is over both $s[n]$, $\forall n$, and the channel.

3) Outage Probability: The outage probability provides a statistical view of the computation error as compared with MSE. By using (12), it can be defined as the probability that the normalized FEE is larger than or equal to $\epsilon$ [39], [66], i.e.,

\[
P_{\text{out}}(\epsilon) = \mathbb{P}(\epsilon_{\text{FEE}}(\hat{f}(s[n])) \geq \epsilon).
\]

Based on [4], for a given $\epsilon$, the block outage probability may also be defined by

\[
P_{\text{out}}(\epsilon) = \mathbb{P}\left(\bigcup_{n=1}^{N_f} \sup_{s[n]} |\hat{f}(s[n]) - f(s[n])| > \epsilon\right).
\]
4) **Computation Error Rate:** computation error rate (CER) is analogous to the bit error rate in communication systems. It can be used for assessing the computation when the arguments of the functions are discrete values. Let \( s_k[n] \) be a discrete random variable, \( \forall k, n \). The CER can be defined as

\[
P_{\text{cer}} \triangleq \Pr \left( \hat{f}(s[n]) \neq f(s[n]) \right).
\]

If \( N_f \) functions are taken into account for the error-rate calculation, as done in [3] and [69], the block-computation error rate (BCER) can be expressed as

\[
P_{\text{bcer}} \triangleq \Pr \left( \bigcup_{n=1}^{N_f} \hat{f}(s[n]) \neq f(s[n]) \right).
\]

5) **Computation Rate and Throughput:** Computation rate \( \mathcal{R} \) and computation throughput \( R \) can be defined as the number of functions computed per channel use [3], [4], [69] and the number of functions computed per second, respectively. They can be respectively expressed as

\[
\mathcal{R} = \frac{N_f}{D} \text{ [functions/dimension]},
\]

and

\[
R = \frac{N_f}{T} \text{ [functions/second]},
\]

where \( D = 2N_{\text{access}}N_f \) and \( T \) are the number of real dimensions and the interval used for computing \( N_f \) functions.\(^{4}\)

It is worth noting that the number of real dimensions for a single-input-single-output scenario is approximately equal to \( 2WT \) for given bandwidth \( W \) and time interval \( T \) [70, Sec. 4.6]. Hence, for \( S \) spatial streams, the computation throughput can be approximately calculated as \( R \approx 2WS \times \mathcal{R} \).

Based on [3] and [69], the achievable computation rate can be defined as follows:

**Definition 5 (Achievable Computation Rate [3], [69]):** Let \( s_k[n] \) be a discrete random variable, \( \forall k, n \). The rate \( R \) is said to be achievable if there exists a sequence of length-\( B \) block codes such that \( \lim_{B \to \infty} P_{\text{bcer}} = 0 \).

In [7], the computation of modulo-\( p \) sum function, i.e.,

\[
f(s[n]) = \sum_{k=1}^{N_f} q_k s_k[n], \forall n,
\]

for \( s_k[n], q_k \in \mathbb{F}_p \), is investigated over Gaussian channel, i.e., \( \epsilon_b = \sum_{k=1}^{K} h_k \epsilon_{k,b} + n_b \) (see Fig. 2). For \( n_b \in \mathbb{R} \), in [7, Th. 2], Nazer and Gastpar show that achievable computation rate is

\[
\mathcal{R} < \mathcal{R}_{\text{comp}} = \frac{\frac{1}{2} \log_2 \left( \frac{1}{2} \left( \left| a \right|^2 \left| b \right|^2 - \frac{P}{2} \right) \right)}{\log_2 (p)},
\]

where \( a = [a_1, \ldots, a_k] \in \mathbb{Z}^K \) for \( q_k = g^{-1}(a_k \mod p) \), i.e., \( g^{-1} \) is a map from \( \{0, 1, \ldots, p - 1\} \) to the corresponding elements in \( \mathbb{F}_p \), and \( P \) is the average transmit power, \( h = [h_1, \ldots, h_K] \in \mathbb{R}^K \). By observing the fact that, for a sufficiently large prime \( q \), computing the modulo-\( q \) sum of \( K \) elements for \( s_k[n] < p \) is equivalent to the arithmetic sum for \( q > (p - 1)K \) and using [1, Th. 1] and (20) for \( a_k = 1, \forall k \).

Jeon et al. in [3] show that the achievable computation rate can be expressed as

\[
\mathcal{R} < \mathcal{R}_{\text{comp}}(f, \epsilon) = \frac{\frac{1}{2} \log_2 \left( \frac{1}{2} \left( \left| a \right|^2 \left| b \right|^2 - \frac{P}{2} \right) \right)}{\log_2 ((2b_0(f, \epsilon))^{-1} + \log_2 K)},
\]

where \( b_0(f, \epsilon) \) is the number of bits defined in Section III-C2a for given function \( f \) and the amount of maximum distortion \( \epsilon \).

In Fig. 3, the achievable computation rates for a given number of EDs based on the example given in [3] for arithmetic sum. Suppose the symbols \( s_k[n], \forall k, n \), are independent and identically distributed (IID) Bernoulli random variables with the probability \( 1/2 \) and the signal-to-noise ratio (SNR) is 15 dB. From [71], \( H(f(s[n])) \) can be calculated as

\[
H(f(s[n])) = K - 2^{-K} \sum_{K=1}^{K} (\frac{K}{2}) \log_2 (\frac{K}{2}) \leq \frac{1}{2} \log_2 (\pi e K/2).
\]

Also, \( b_0(f, \epsilon) = 1 \) is sufficient for the representation of the symbols. The achievable computation rate for the separation of communication and computation can be calculated as

\[
\frac{1}{\mathcal{K}} \log_2 (1 + K/\sigma_n^2) [3], [72, eq. (6.13)].
\]

As can be seen from Fig. 3, the achievable computation rate with the separation of communication and computation can be dramatically lower than the ones with joint communication and computation. For example, if there are \( K = 100 \)
TABLE II
COMPUTATION UNDER FADING CHANNEL

| Category | Method | $N_i$ | $N_c$ | Structure of channel $H_k$ | Linear precoder $B_k$ | Linear decoder $A$ | Decode (4) | Rate $R$ (4/5) | $x_k$ | Calculated function | Target function |
|----------|--------|-------|-------|---------------------------|---------------------|-----------------|------------|-----------------|------|------------------|---------------|
| CSIT     | ZF     | $M$   | $L$   | $[h_{k,1}, \ldots, h_{k,L}]^T$ | $1$                | $\frac{1}{M}\sum_{k=1}^K h_k$ | $\{x_k\}$ | $y = \sum_{k=1}^K H_k x_k + n = \sum_{k=1}^K x_k + n$ | $\frac{1}{M} \sum_{k=1}^K h_k$ | $\frac{1}{M}$ | $\sum_{k=1}^K x_k$ |
| CSIT     | MMSE   | $M$   | $L$   | $[h_{k,1}, \ldots, h_{k,L}]^T$ | $1$                | $\frac{1}{M}\sum_{k=1}^K h_k$ | $\{x_k\}$ | $y = \sum_{k=1}^K H_k x_k + n = \sum_{k=1}^K x_k + n$ | $\frac{1}{M} \sum_{k=1}^K h_k$ | $\frac{1}{M}$ | $\sum_{k=1}^K x_k$ |
| CSIT     | Channel hardening via aggregated CSI | $M$ | $L$ | $[h_{k,1}, \ldots, h_{k,L}]^T$ | $1$                | $\frac{1}{M}\sum_{k=1}^K h_k$ | $\{x_k\}$ | $y = \sum_{k=1}^K H_k x_k + n = \sum_{k=1}^K x_k + n$ | $\frac{1}{M} \sum_{k=1}^K h_k$ | $\frac{1}{M}$ | $\sum_{k=1}^K x_k$ |
| CSIT     | N × M  | $M$   | $L$   | $[h_{k,1}, \ldots, h_{k,M}]^T$ | $1$                | $\frac{1}{M}\sum_{k=1}^K h_k$ | $\{x_k\}$ | $y = \sum_{k=1}^K H_k x_k + n = \sum_{k=1}^K x_k + n$ | $\frac{1}{M} \sum_{k=1}^K h_k$ | $\frac{1}{M}$ | $\sum_{k=1}^K x_k$ |

EDs, OAC promises approximately 10 times faster reliable computations than the one with separation. For this example, Goldenbaum’s result is more conservative than Jeon’s rate as Goldenbaum’s expression implicitly assumes equiprobable final outcomes. In [42, p. 4], for the same scenario, it is argued that the EDs can send significantly less information to the ES if the objective is to compute the sum function. This is because the entropy of the computed function is much smaller than the amount of information that needs to be transferred to the ES with the separation, i.e.,

$$H(f(s[n])) \leq \sum_{k=1}^K H(s_k[n]) = K.$$  

In [69] and [73], to improve the computation rate, a construction of the target function via local functions is proposed. In [69], the authors assume that the channel coefficients are IID random variables and they select a subset of EDs with the largest channel gains to gradually calculate the target function over fast-fading channels. It is shown that a non-vanishing computation rate is achievable even if the number of EDs in the network increases. In a recent work [74], nested lattice codes are utilized along with stochastic quantization for the AWGN channel. A theoretical comparison between uncoded OAC, coded OAC, and separation of communication and computation is provided in terms of distortion under the AWGN channel. In [75], the computation rates for both homogeneous and heterogeneous networks under large-scale fading are studied. The reader is also referred to [17] for the computation rates for uncoded and coded physical layer network coding strategies.

B. Classification Criteria: Availability of CSI

One of the challenges for calculating (1) with an OAC method arises due to the fact that the impact of multi-path distortion on the transmitted symbols from the EDs (i.e., $H_k x_k$) occurs before the superposition, as expressed in (8). Hence, an estimator that estimates the output of the function at the ES can be unrealized, i.e., it may not be written solely as a function of the received symbols. For example, consider a scenario where $H_k$ is IID and follows Rayleigh fading and the power control ensures that the average received signal powers are perfectly aligned, e.g., $P_k = 1, \forall k$. Since the superposed symbol at the ES in this case cannot be expressed as

$$y = \sum_{k=1}^K H_k x_k + n = \sum_{k=1}^K x_k + n,$$

where $H \in \mathbb{C}^{N_i \times N_t}$ is a matrix independent of the vector $x_k$, $\forall k$ (i.e., there is no uniform fading matrix $H$ [67]), an equalizer that relies on the availability of $H$ to obtain $\sum_{k=1}^K x_k$ cannot be realized. Given this observation, in this subsection, we classify the methods in the state-of-the-art based on how they deal with the fading due to the wireless channel through the precoders $B_k, \forall k$, and the decoder $A$. We group them based on the availability of CSI at the transmitter (CSIT) and CSI at the receiver (CSIR). For simplifying the classification, we assume $P_k = 1, \forall k$. We also provide corresponding equations for different strategies in Table II when the real part of the superposed symbol is used for computation.

1) CSIT: Available, CSIR: Not Available: In this category, we assume that all EDs have their own CSI (i.e., the $k$th ED knows $H_k$, but not the set $\{H_{k'}|k' \neq k\}$) and the CSI is not available to the ES, i.e., no CSIR. Hence, the EDs cannot make coordinated transmissions. However, each ED can pre-distort its own transmitted symbols under an average transmit power constraint or an instantaneous transmit power constraint by designing the precoders, i.e., $B_k, \forall k$, such that the superposed symbol at the receiver is a good approximation to the desired value. In this category, the equalization used in traditional communication is shifted from the receiver to the transmitter. The computation rate of these schemes can be as high as

$$y = \sum_{k=1}^K H_k x_k + n = \sum_{k=1}^K x_k + n.$$
that results in a non-negative value and afterward the square root of the resulting value is multiplied with a sequence of length $N$ as $\sqrt{g(p_k)} \times [e^{j\theta_1}, \ldots, e^{j\theta_N}]$. At the receiver, the energy of the received sequence is calculated to achieve the OAC and the superposed symbol is processed with another affine function $h$ to reverse the impact of $g$ on the superposed symbols (see Table II for the final expression and Section III-C1b). Three interesting observations were made: 1) Only amplitude correction is needed as the receiver calculates the energy of the symbol. This implies that the EDs need only modulus CSI [103, Proposition 1]. 2) The sequences for the proposed scheme should be designed to harness interference as a common goal, instead of eliminating the interference as in a traditional code division multiple access systems [37]. The need for a sequence set that should satisfy the property of mutually-orthogonal complementary codes or Z-complementary code set in general (e.g., [104, eq. (2)]). In [39], unimodular sequences with random phases are adopted to reduce the interference. 3) As the ES calculates the energy of the received sequence, the proposed scheme is not sensitive to time and phase synchronization errors (see [39, Fig. 2] for an illustration). By following Goldenbaum’s work [39], in [66], it is assumed that the range space of each pre-processing function is a compact subset of non-negative reals. At the EDs, amplitude correction with truncation is applied to the output of the pre-processing function. To ensure both peak and average transmit power constraints, a sequential symbol power adaptation strategy that uses only the observed channel coefficients is proposed. At the ES, the energy of the superposed sequence is calculated to compute the nomographic function. Note that truncation is also used in [105] to realize Goldenbaum and Stańczak’s scheme in practice.

d) Maximum-ratio transmission without CSIR: With maximum-ratio transmission (MRT) for traditional communications, a symbol is transmitted on the strongest eigenmode of the channel matrix and the received signals are combined using maximum-ratio combining (MRC), which requires both CSIT and CSIR [106]. In [107], MRT without CSIR is proposed for a scenario with single-antenna ES and EDs. In this approach, the symbols are multiplied with the conjugate of the channel coefficients. Hence, the transmit power can be more effectively utilized as compared to that of PCs if the symbols at the ED observe channel coefficients with different magnitudes. However, the norm-square of the channel appears on the calculated function (see Table II). The power control factor for this approach can be designed based on average transmit power (i.e., $\eta_k = 1/\|\mathbf{h}_k\|^2_2/N_t$ or instantaneous transmit power (i.e., $\eta_k = 1/\|\mathbf{h}_k\|^2_2$ for $N_t > 1$) [108].

e) What can go wrong?: The methods in this category requires accurate CSITs at the EDs. However, as discussed in [39], this can impose stringent requirements on the underlying mechanisms such as time and phase synchronizations, channel estimation, and channel prediction, which can be challenging to satisfy without a clock synchronization and/or under the non-stationary channel conditions in mobile wireless networks [109], [110]. A sample-level precise synchronization (i.e., not just within the cyclic prefix (CP) duration of an
orthogonal frequency division multiplexing (OFDM) symbol) is needed for the methods that are sensitive to the phase distortion as the CSI is a function of the synchronization point at the EDs and the time-of-arrivals of the transmitted signals from the EDs. The second challenge is that it is not trivial to extend these methods to the cases with \( N_t > N_t \) (e.g., an ES with multiple antennas or a multi-cell computation) [111]. This is due to the fact that the precoder cannot achieve a channel inversion for a random channel matrix without interference (i.e., \( \mathbf{H}_k \mathbf{B}_k \neq \mathbf{I}_{N_t} \) for \( N_t > N_t \)). One potential solution to this issue is coordination among the EDs (e.g., through some orchestration by an ES or multiple ESs) as done in uniform forcing, which unfortunately requires CSIR.

2) CSIT: Available, CSIR: Available: In this category, we assume that the CSI is available at both EDs and ESs. Hence, it is the most flexible framework for optimizing the OAC performance as the precoders, i.e., \( \mathbf{B}_k, \forall k \), and the decoder, i.e., \( \mathbf{A} \), can be designed jointly.

a) ZF and MMSE coordinations: For ZF coordination [64], [67], [112], [113], [114], the noise at the ES is ignored. In this case, if \( \mathbf{A}^{H} \mathbf{H}_k \mathbf{A} \) is an invertible matrix, \( \forall k \), for a given \( \mathbf{A} \approx \mathbf{A}_n / \sqrt{\eta} \) for any \( \mathbf{A}_n \in \mathbb{C}^{L \times N_t} \) such that \( \text{tr}(\mathbf{A}_n^{H} \mathbf{A}_n) = L \), the precoder \( \mathbf{B}_k, \forall k \), can be obtained as

\[
\mathbf{B}_k = (\mathbf{A}^{H} \mathbf{H}_k)^{\dagger} = \sqrt{\eta}(\mathbf{A}^{H} \mathbf{H}_k)^{H}(\mathbf{A}_n^{H} \mathbf{H}_k \mathbf{H}_k^{H} \mathbf{A}_n)^{-1},
\]

where \( \max_k \text{tr}(\mathbf{B}_k \mathbf{B}_k^{H}) \leq P_0 \) must hold true for a given maximum transmit power \( P_0 \). By evaluating the condition further, the power control factor \( \eta \) can be obtained as

\[
\eta = \frac{P_0}{\min_k \text{tr}(\mathbf{A}_n^{H} \mathbf{H}_k \mathbf{H}_k^{H} \mathbf{A}_n)^{-1}}.
\]

Hence, the corresponding MSE of the superposed modulation symbols can be expressed as

\[
e(\eta, \mathbf{A}_n) \triangleq \mathbb{E}\left\{ \left\| \mathbf{m} - \sum_{k=1}^{K} \mathbf{m}_k \right\|_2^2 \right\} = \sigma_n^2 \frac{\text{tr}(\mathbf{A}_n^{H} \mathbf{A}_n)}{\eta}.
\]

By substituting \( \eta \) into (27), the optimization problem for ZF coordination can be written by

\[
\mathbf{A}_n^{*} = \arg \max_{\mathbf{A}_n} \min_k \text{tr}(\mathbf{A}_n^{H} \mathbf{H}_k \mathbf{H}_k^{H} \mathbf{A}_n)^{-1}
\]

s.t. \( \text{tr}(\mathbf{A}_n^{H} \mathbf{A}_n) = L \).

Since \( \mathbf{A}^{H} \mathbf{H}_k \mathbf{H}_k^{H} \mathbf{A} \) is assumed to be an invertible matrix, \( \forall k \), the maximum number of computable functions is \( L \leq \min\{N, M\} \). The computation rate can be calculated as \( L/M \) if both real and imaginary parts of the symbols are used for computation. If \( \mathbf{B}_k \) and \( \mathbf{A} \) represent the precoder and decoder for a multi-antenna system, respectively, \( S = L \) spatial streams via multiple antennas can be utilized to compute \( L \) functions in parallel, i.e., a higher computation throughput (see the definition in Section III-A5). Also, note that the design problem for OAC is the dual of the beamforming optimization for the downlink (DL) multi-casting [67], [112].

In [67], [113], [114], ZF coordination is investigated for a scenario with single-antenna EDs and a multiple-antenna ES. In [67], an algorithmic approach based on semidefinite relaxation is considered to solve (28). In [113] and [114], ZF-coordination is studied based on device subset selection. In [114, Th. 1], a closed-form solution is provided when only a single ED transmits with the maximum power under certain channel conditions. The case where both EDs and ES have multiple antennas is studied in [64] and [112]. In [64], ZF coordination is investigated for \( \mathbf{A}_n^{*} = \mathbf{I}_L \) and \( N_t \geq N_t = L \). An approximate solution for \( \mathbf{A}_n^{H} \mathbf{A}_n = \mathbf{I}_L \) is provided in [112, eq. (14)].

With minimum mean-squared error (MMSE) coordination [68], [114], [115], [116], [117], the main goal is to minimize the MSE of the superposed modulation symbol vector \( \mathbf{m} \), where the MSE can be written as a function of \( \mathbf{A} \) and \( \mathbf{B}_k, \forall k \), by

\[
e(\mathbf{A}, \{\mathbf{B}_k\}) \triangleq \mathbb{E}\left\{ \left\| \mathbf{m} - \sum_{k=1}^{K} \mathbf{m}_k \right\|_2^2 \right\} = \mathbb{E}\left\{ \left\| \sum_{k=1}^{K} (\mathbf{A}^{H} \mathbf{H}_k \mathbf{B}_k - \mathbf{I}_L) \mathbf{m}_k + \mathbf{A}^{H} \mathbf{H}_n \right\|_2^2 \right\}
\]

\[
= \sum_{k=1}^{K} \text{tr}\left\{ (\mathbf{A}^{H} \mathbf{H}_k \mathbf{B}_k - \mathbf{I}_L)(\mathbf{A}^{H} \mathbf{H}_k \mathbf{B}_k - \mathbf{I}_L)^{H} \right\}
\]

\[
+ \sigma_n^2 \text{tr}\{\mathbf{A}^{H} \mathbf{A}\},
\]

for \( \mathbb{E}\left\{ \mathbf{m}_i \mathbf{m}_j^{H} \right\} = \delta_{ij} \mathbf{I}_L, \forall i, j \). For MMSE coordination, the optimization problem can be expressed as

\[
(\mathbf{A}^{*}, \{\mathbf{B}_k^{*}\}) = \arg \min_{\mathbf{A}, \{\mathbf{B}_k\}} e(\mathbf{A}, \{\mathbf{B}_k\})
\]

s.t. \( \text{tr}(\mathbf{B}_k^{H} \mathbf{B}_k) \leq P_0, \forall k \)

The MMSE coordination differs from the ZF coordination in that it leads to a combination of maximum power transmission and channel inversion across the EDs, and the optimal precoders and decoder are functions of the noise variance at the ES. In [115] and [118], the authors investigate MMSE-coordination for a scenario with single-antenna ES and EDs. Reference [115] extends the optimization problem to time-varying channels while [118] investigates the effect of imperfect CSI on the MMSE computation and provides several closed-form solutions based on some approximations. In [68] and [114], a scenario where the EDs are equipped with a single antenna while ES has multiple antennas is considered. It is shown that the MMSE coordination is related to the device subset selection problem. In [116], a single function, i.e., \( L = 1 \), is aim to be computed in multiple-input-multiple-output (MIMO) channel and an iterative algorithm described in [116, eqs. (11)–(13)] is adopted for MMSE coordination. In [119], MMSE coordination is considered by taking the normalizations of the parameters into account for distributed sensing.

In the literature, the optimization of the aforementioned linear precoders and decoder have been investigated for various interesting applications and scenarios. In [120], analog beamforming is investigated for OAC, where the main objective is to minimize a bound on the loss function for FEEL, instead of MSE. In [121], the scenario is extended to a multi-cluster network and uniform forcing is investigated under inter-cluster interference in selective fading. In [122], the receive beamforming is optimized based on antenna selection. In [123],...
the same scenario is investigated with the consideration of a relay network with multiple antennas and the beamforming vectors at the EDs, ESs, and relays are jointly optimized. To calculate multiple linear functions, the design of beamforming vectors at the ES and ED is discussed in [124]. In [125], a general distribution optimization problem is investigated when the radios are equipped with a large number of antennas for transmission and full-duplex capability. Beamforming vectors are proposed to be optimized to support multiple concurrent computations. In [126], several channel inversion techniques along with scheduling are investigated when there are multiple antennas at the ES. For a given maximum tolerable computation error [126] or FEEL performance [127], greedy scheduling algorithms are proposed. In [128], the authors design the precoders at the EDs and the decoder at the ES with the considerations of both spatial correlation and heterogeneous data correlation to minimize MSE further. In [129], simultaneous federated learning with OAC and information transmission is studied.

b) Diversity-oriented techniques: In the literature, some methods exploit time and/or frequency diversity techniques to improve the performance of OAC based on pre-equalization. For example, in [130], a WSN scenario, where each sensor sends a symbol over multiple subcarriers, is considered. The pre-equalization vector at the sensors and the aggregation vector that combines the copies at the fusion center are jointly optimized such that MSE is minimized under a power constraint. The utilized precoder and combining vectors are similar to the ones with multiple antennas [113] since the effective channel can be expressed as a diagonal MIMO matrix. In [131], a space-time approach for multiple EDs and ESs is also investigated to minimize the MSE of the computation. Similarly, in [132], a multi-slot OAC is proposed for fast-fading channels and time diversity is exploited to mitigate the impact of fading channels on OAC.

In [133], an OAC strategy using a space-time line code (STLC) [108] is proposed to achieve a receive diversity gain. In this approach, each ED is equipped with a single antenna while the ES has two antennas. Each ED performs two STLC symbol transmissions back-to-back as a function of the CSI, where STLC symbols are generated from the same sensor information. The ES combines the received symbols at different time slots and antennas blindly. Nevertheless, the power normalization factor still needs to be known at the EDs, where the ES calculates the factor based on the feedback transmitted from all EDs in orthogonal channels.

c) Channel manipulation: One way of achieving favorable propagation conditions for OAC is to manipulate the multi-path channel itself with technologies like reconfigurable intelligent surfaces (RIS) [134]. For example, in [135], RIS is utilized to boost or null the received signal power at desired locations. In [136], device selection for OAC is studied along with RIS with the availability of CSIT. In [137], the RSI phase shifts are exploited for over-the-air model aggregation with the consideration of the cascaded channel coefficients to eliminate the need for CSI at the EDs. In [138], multiple RISs are investigated for a similar scenario. In [139] and [140], the authors consider the optimization problem of transceiver design and RIS phase selection with the consideration of imperfect CSI. In [141], the sign stochastic gradient descent (signSGD) is exploited for OAC along RIS. In [142], average MSE with respect to a target function is minimized by jointly optimizing the RIS phase-shift vector and the transmission and reception scaling factors of ED. In [143], RIS is proposed to compute convolution over the air to realize a convolution neural network (CNN) in the wireless channel. In [144], the authors investigate a joint optimization problem concerning the transmit power, denoising factor, and RIS phase-shifts for a graph neural network. We refer the reader to [145] and the references therein for further optimization frameworks on phase shift design.

d) What can go wrong?: The methods in this category can introduce a major computation complexity to both EDs and ES. They are also prone to imperfections caused by underlying enabling mechanisms, e.g., imperfect coordination among EDs within the coherence time, and phase, time, and frequency synchronization errors. Hence, the OAC performance of these methods is a strong function of how much the underlying link-layer procedures in practice can make the assumptions (e.g., accurate and fresh CSI estimates at both ES and ED within the coherence time) hold.

3) CSIT: Not Available, CSIR: Available: This category is dedicated to blind EDs, i.e., EDs cannot access the CSI, but the ES has some knowledge about the CSI. The methods in this category particularly rely on channel hardening techniques.

a) Channel hardening by using the aggregated CSI: One way of achieving channel hardening for OAC is to use an estimate of the superposed channel, i.e., the sum of the channel gains from all the EDs (i.e., \( \sum_{k} h_{k} \)) to derive the linear decoder \( A \) at the ES [146], [147], [148], [149], [150]. Since the CSI between each ED and ES is not needed with this strategy, the channel estimation overhead significantly reduces at the expense of some interference due to the uncoordinated transmissions of the EDs. In [146], [147], this approach is adopted based on multiple antennas at the ES. It is assumed that the ES has a noisy estimate of the aggregated channel from all the devices to each antenna and employs MRC. It is shown that the variance of interference on the superposed symbol is scaled by \((K - 1)/N_{t}\), where \( N_{t} \) is the number of antennas at the ES [146], [147]. In [151], the same scenario is investigated for time-varying channels and it is shown that the time-variation in typical wireless channels does not reduce the FEEL performance and the inter-carrier interference reduces with the increasing number of receive antennas.

b) Advanced receivers: In [152], [153], the digital OAC problem is interpreted as a multi-user detection problem. Considering an asynchronous multi-user OFDM scenario, it is demonstrated that multi-user detection algorithms can be applied to the superposed signals for convolution code or low-density parity check (LDPC). In [154], a similar multi-user detection and aggregation approach is considered for Long-Range (LoRa) networks. In [155], it is proposed to separate the transmitted signal of each client from the superposed signals so that independent sparsification patterns can be applied at the EDs by assuming that the number of antennas is larger than the number of EDs.
c) What can go wrong?: Channel hardening relies on the existence of a large number of degrees-of-freedom (DoF) at the ES, i.e., $N_{t}$, to decrease the variance of interference, which can increase the cost and decrease the computation rate $R$ as can be seen from Table II. Also, if $N_{t}$ is larger than or equal to $KN_{t}$ and the complete CSIR is available at the ES, the separation of computation and communication tasks can be more reliable than the OAC as the number of observations can be larger than or equal to the number of unknowns. The main limitation of advanced receivers for the detection of codewords from the superposed signals is that it is not trivial to extend the computation to a large number of EDs as the computation and storage complexity can be prohibitively high.

4) CSIT: Not Available, CSIR: Not Available: The methods in this category is the most restrictive in the sense of optimization as the CSI is available neither at the EDs nor the ES. Although this appears as a prohibiting factor for a reliable OAC, the main benefits gained by not using CSI are the robustness against time-variation of the wireless channel and synchronization errors and a major overhead reduction as compared to the methods relying on the availability of CSIT or CSIR.

a) Orthogonal signaling: In [156], [157], [158], [159], the authors consider distributed training by the MV with signSGD [160] and calculate the MV based on orthogonal signaling at the EDs and non-coherent detection at the ES. Since the arguments and the output of the nomographic function in this specific application consist of discrete states, i.e., $\{-1, 1\}$, the schemes in [156], [157], [158], [161], and [162] use frequency-shift keying (FSK) over OFDM, pulse-position modulation (PPM), and chirp-shift keying (CSK) over discrete Fourier transform (DFT)-spread OFDM (DFT-s-OFDM), where the symbols are determined based on $\text{sign}(s_k)$. An energy comparator is used to detect the MV at the ES. The authors show the efficacy of this approach for distributed learning under time synchronization errors without requiring phase synchronization (see [163] for demonstration). Since these schemes do not use CSIT and CSIR, they can also be used for multi-cell computation, where there are multiple fusion nodes [159]. The authors in [164] also propose to utilize orthogonal resources for negative and positive-valued measurements. As compared to [164], in [165], multiple orthogonal resources are used for data encoding while [162] uses them for multiple MV computations. Note that the distributed training based on MV with signSGD is considered in [78] and [141], where the OAC relies on TCI discussed in Section III-B1a and the use of RIS, respectively. signSGD along with error feedback is investigated in [166]. However, phase synchronization is needed for these methods as [78], [141], [166] use the same resource for negative and positive gradients.

The keying idea is generalized in [167], [168] to compute general nomographic functions by using balanced numerals. With similar motivations, i.e., robustness against synchronization errors, Valenti and Ferrent study FSK for physical-layer network coding (PLNC) in [169], [170], where the target function is the XOR of the transmitted bits from two devices. In [171], [172], it is proposed to project the gradient vector onto a set of orthogonal basis. In [173], a positional encoding for OAC without using CSI is discussed. Based on the fraction of time slots occupied among a fixed number of available slots, the activated classes that encode temperature ranges are estimated. In [174], a codebook for TBMA is proposed to be designed via a neural network with the considerations of channel and source statistics.

b) Channel hardening and energy estimation: In [103], Goldenbaum and Stańczak re-evaluate their scheme presented in [39] when there is no CSIT under a scenario where the receiver is equipped with multiple antennas. In this approach, each ED transmits a sequence as discussed in Section III-B1c. However, it does not apply any amplitude correction and the receiver at the ES calculates the energy of the received sequence over the multiple antennas. One of the main conclusions drawn from this approach is that the rapid changes of the channel coefficients can be beneficial to improve the convergence when CSIT is not available for IID small-fading. Note that a similar approach is also adopted in more recent work in [57], [175], [176], and [54, Sec. VI]. To mitigate the channel fading coefficients on OAC, averaging over multiple antennas is investigated in [167], [177]. The impact of an uncompensated channel along with multiple transmissions is discussed for collaborative-decision making in [97].

c) Joint channel and parameter estimation: In [178], the blind OAC is interpreted as a joint channel and parameter estimation problem. To estimate the channel coefficients and parameters, a randomly initialized Wirtinger flow is proposed. It is demonstrated that the proposed approach results in small estimation errors with sufficient samples.

4) What can go wrong?: The error rate or MSE with orthogonal signaling can be worse than the methods relying on pre-equalization or multi-antenna techniques as the impact of the channel on the symbol is not compensated. To reduce the estimation errors, averaging can consume time, frequency, or space resources (i.e., a lower computation rate) while introducing a complex algorithm to the ES can increase the receiver complexity.

C. Classification Criteria: Encoding

The classification in this category particularly concerns how the outputs of the pre-processing functions are processed before the linear encoders take place to overcome the fading channel. We group the approaches under analog encoding and digital encoding. While analog encoding deals with the continuous-valued symbols to realize the desired nomographic function (over an analog modulation), digital encoding uses some form of quantization, compression, and/or source-channel coding for the same goal (over a digital modulation).

1) Analog Encoding: In the literature, a majority of the OAC methods exploit the available DoF without a particular encoding as long as the reliable superposition is maintained under the fading channel with aid of the precoders $\mathbf{B}_k$, $\forall k$, and the decoder $\mathbf{A}$. Nevertheless, it has been shown that additional processing can still be helpful for certain goals.
a) Linear analog encoders: A linear analog encoder for OAC can be defined as
\[ \epsilon_k \{ p_k \} = G p_k, \forall k, \] (30)
where \( G \) is a \( B \times N_t \) matrix for \( B < N_t \) (i.e., compression) [179]. Hence, the encoder projects the vector \( p_k \) into a lower dimensional space and reduces the number of resources to be utilized to compute \( N_t \). A linear encoder is a suitable operation for OAC because the sum of the projected vectors is identical to the projection of the sum of vectors.

Linear encoders are especially applied to applications where the pre-processed symbol vector \( p_k \) is inherently sparse or can be sparsified with a tolerable distortion. For example, in [180] and [77], for distributed learning, the EDs sparsify their gradient estimates before they project them into a lower dimensional space with (30). These projections are directly used with an analog OAC scheme. At the ES, an approximate message passing algorithm is proposed to recover the superposed symbols. In [179], with a similar motivation, \( G \) is constructed randomly from the rows of the rotated Walsh-Hadamard matrix. At the ES, the projected vector is mapped to the superposed symbol based on a general norm-minimization problem, i.e., \( p = \min_k \| e - G p \|_2 \). It is shown that linear analog encoders in the AWGN channel can perform well at low SNRs. In [181], the sparsification is achieved by selecting \( k \) gradients with the greatest magnitudes, where \( k \) is a predetermined integer. In this study, a discrete cosine transform matrix is used for the compression. In [155], a similar compression approach is considered for the model updates and the measurement matrix is constructed based on IID Gaussian distribution.

One of the concerns on OAC with sparsification is that the sum of the sparse vectors may not be sparse after the aggregation, i.e., the sparsity level can deteriorate depending on the number of EDs. In [155], the authors study a uniform sparsity pattern and independent sparsity patterns for FEEL and the references therein) is utilized to map the neural network parameters onto continuous-values on Archimedes’ spirals for unequal error protection and improving bandwidth efficiency. Although the proposed method is not used for OAC, introducing similar analog coding approaches to OAC is an unexplored area and may improve the reliability of the computation.

2) Digital Encoding: In this subsection, we discuss the methods that use some form of source and channel coding for OAC.

a) Nested lattice codes: The linearity of the nested lattice codes are first used in [1] and [7] for achieving computation over Gaussian channels in finite fields (see Section III-A5 for further discussions). In [4] and [189], a digital OAC scheme along with a nested lattice coding is introduced to increase the reliability of the computation of \( N_t \) functions in a real-valued AWGN channel. To calculate \( \sum_{k=1}^{N_t} p_k[n] \), \( \forall n \), the steps taken at the \( k \)th ED, \( \forall k \), based on [4, Th. 5], can be listed as follows:

- **Step 1 (Quantization):** Let \( b_0 \) be the quantization parameter in bits, which is specified based on the amount of tolerable quantization error. The symbol \( p_k[n] \), \( \forall n \), is mapped to a positive integer \( w_k[n] \in \{0, 1, \ldots, 2^{b_0} - 1\} \) for a given \( b_0 \). For example, for \( b_0 = 3 \) bits, \( K = 3 \) EDs, and \( N_t = 4 \) functions to be computed, suppose that the EDs’ symbols within the range \([0, 1]\) are given by

\[
\begin{align*}
p_k[n] & = 1 & n &= 1 & n &= 2 & n &= 3 & n &= 4 \\
k &= 1 & k &= 1 & k &= 1 & k &= 0 & k &= 0.4 \\
k &= 2 & k &= 1 & k &= 1 & k &= 0.7 & k &= 0.7 \\
k &= 3 & k &= 1 & k &= 1 & k &= 0.2 & k &= 0.1 \\
\end{align*}
\] (34)

By uniformly dividing the range \([0, 1]\) into \(2^3\) parts and using a natural code, the corresponding integers can be obtained as

\[
\begin{align*}
w_k[n] & = 1 & n &= 1 & n &= 2 & n &= 3 & n &= 4 \\
k &= 1 & k &= 7 & k &= 7 & k &= 6 & k &= 3 \\
k &= 2 & k &= 7 & k &= 7 & k &= 0 & k &= 5 \\
k &= 3 & k &= 7 & k &= 7 & k &= 1 & k &= 1 \\
\end{align*}
\] (35)

- **Step 2 (Source Encoder):** The integers obtained from the quantization step, i.e., \( w_k[n], \forall n \), are divided into \( S \) sub-sequences and each sub-sequence is mapped to a message as an integer. The procedure is as follows: Each sub-sequence contains \( \tau = N_t/S \) integers for a given \( S \). The \( \tau \) integers in each sub-sequence is first assigned to the digits of a number, e.g., \( w_k[\tau] \cdots w_k[2]w_k[1] \).

The fundamental reason for the square-root operation is that the OAC relies on the estimation of the energy of the sequence at the receiver.
in the base-$\beta$ positional numeral system for $\beta = K(2^{b_0} - 1) + 1$. Afterward, each number in base $\beta$ is converted to an integer, e.g., $\sum_{i=1}^{t} w_i[t]^{\beta-1}$. Note that the source encoding results in $S$ messages, where the maximum value of a message is $(\beta^r - 1)/K$. The reason for choosing $\beta$ as a function of number of EDs is to eliminate a potential carry digit for the superposed message across $K$ EDs so that $\sum_{i=1}^{K} w_i[\tau] \cdot \sum_{i=1}^{K} w_i[2] \cdot \sum_{i=1}^{K} w_i[1] \beta$ can be expressed as $(\sum_{i=1}^{K} w_i[\tau] \cdot \sum_{i=1}^{K} w_i[2] \cdot \sum_{i=1}^{K} w_i[1]) \beta$. For example, consider the integers in (35). Suppose that the source encoder results in $2$ messages across $K$, and the decoder results in $483$ and $163$ as the superposed messages in (37), and the decoder can be calculated over $\tau = N/2 = 2$ integers by mapping them to the digits of a number in a positional numeral system. The corresponding numbers are then converted to the messages as

$$
\begin{align*}
\text{From base } \beta & \text{ to base } 10 \\
\begin{array}{cccccccc}
\beta & 1 & 2 & 3 \\
77 & 63 & 11 & 135 & 2161 & 15 & 36123
\end{array}
\end{align*}
$$

for $\beta = K(2^{b_0} - 1) + 1 = 22$. Hence, the sum of the messages across $K = 3$ EDs can be calculated without a carry digit, e.g., the sum $(77)_{22} + (77)_{22} + (77)_{22} = (21, 21)_{22} \equiv (483)_{10}$ for $n = 1$. Note that the maximum value of the superposed message for this example is $\beta^r - 1 = 483$ while the maximum value of a message is $(\beta^r - 1)/K = 161$ as can be seen from (36).

- **Step 3 (Channel Encoder):** A nested lattice code from $\mathbb{Z}_P^S$ to $\mathbb{R}^B$ is constructed for a prime $p \geq \beta^r$. The $S$ messages calculated from the previous step are used to calculate the codeword $c_k \in \mathbb{C}^B$ with the corresponding generator matrix $G \in \mathbb{R}^{B \times S}$ of the lattice under an average power constraint. For example, the codewords at EDs for the message in (36) can be calculated as

$$
\begin{align*}
\text{For } k & = 1, 2, 3, c_k \equiv G \cdot [161 \ 135] \quad \text{for } k = 1 \\
& \quad \quad G \cdot [161 \ 5] \quad \text{for } k = 2 \\
& \quad \quad G \cdot [161 \ 23] \quad \text{for } k = 3.
\end{align*}
$$

The ES receives the sum of the codeword, i.e., $\hat{c}$. For example, for the codewords in (37), $\hat{c}$ can be expressed as

$$
\hat{c} = \sum_{k=1}^{3} c_k.
$$

The superposed vector $\hat{c}$ at the ES is processed as follows:

- **Step 1 (Channel Decoder):** By using a Euclidean nearest neighbor decoder, the decoder obtains the message, which corresponds to the superposed message due to the linearity of the code. For example, for the codewords in (37), $\hat{c}$ can be expressed as

$$
\hat{c} = G \cdot [483 \ 163]
$$

and the decoder results in $483$ and $163$ as the superposed messages if there is no error.

- **Step 2 (Source Decoder):** Each superposed message is expressed in base $\beta$ to obtain the sum of the quantization results, e.g., $(\sum_{i=1}^{K} w_i[\tau] \cdot \sum_{i=1}^{K} w_i[2] \cdot \sum_{i=1}^{K} w_i[1]) \beta$. For example, $(483)_{10} \equiv (21, 21)_{22}$ and $(163)_{10} \equiv (79)_{22}$. Hence, the source decoder returns $\hat{p} = [\hat{p}[1], \hat{p}[2], \hat{p}[3], \hat{p}[4]]^T = [21, 21, 7, 9]^T$, i.e., the sums of the values on each column in (36).

**b) Encoding based on numeral systems:** In [190], pulse-amplitude modulation (PAM) is utilized with a radix-based encoding. In this approach, the binary representation of a parameter $p_k$ is first partitioned into subgroups. Afterward, the decimal representation of the bits on each subgroup is mapped to a PAM symbol to achieve a processing gain (see [190, Fig. 4]). This approach generalizes the encoder that encodes $p_k$ into an $M$-PAM symbol after $p_k$ is quantized into $\log_2 M$ bits [179, eq. (13)]. Note that PAM can be extended to square $M$-quadrature amplitude modulation (QAM) constellations, where real and imaginary part of the constellation distributes to two symbols, as done for gradient aggregation for FEEL in [82].

In [13], the authors propose to calculate $\sum_{k=1}^{K} p_k$ where $p_k = w_k s_k$ for non-negative integers $s_k$, $w_k$. They consider the binary representation of each symbol and map each bit to a binary phase shift keying (BPSK) symbol. By using the linearity of the decomposition, the weights $w_k$ are incorporated to the transmit power at the source nodes to calculate the weighted sum. The main observation is that the received symbol is a point in a non-standard PAM after the superposition. The destination node exploits the discrete nature of the superposition to compute the arithmetic sum. The authors also derive the CER (See Section III-A). In [191], the authors investigate the computation problem by mapping the image of the desired function to the discrete points in the constellation after the signal superposition.

In [167], [168], the authors propose to utilize the balanced number systems for OAC to represent a negative parameter efficiently. A balanced numerical system consists of negative numerals, i.e., signed digits. Hence, it can represent a negative number without using a dedicated sign symbol. In this method, the parameters are first represented in a balanced number system. For each numeral, one of the corresponding OFDM subcarriers are activated. At the receivers, the sum of the parameters is estimated. MSE is also analyzed for multiple antennas at the receiver. If only a single digit is used with balanced ternary system, the encoding in [167], [168] corresponds to the keying methods discussed in [156], [157], [158], [159]. The proposed encoding method in [156], [157], [158], [159] is designed for calculating the MV (see Table I). Since the homomorphic function for the MV consist of discrete states, the modulation symbols are determined based on keying methods such as FSK, CSK, and PPM, and non-coherent detectors are used to obtain the MV at the ES. Note that the same homomorphic function is targeted to be computed in [78] based on TCI. In [161], it is shown that using a tri-state encoder (i.e., $\{-1, 0, 1\}$) that eliminates the EDs with small gradients from the MV computation can largely address the bias due to the data heterogeneity and imperfect power control in the learning when OAC is used for FEEL. In [162], CSK is extended to $M$-ary CSK. The sign of $\log_2 M$ parameters are mapped to a chirp index and the authors show how to calculate MVs for this specific encoding by exploiting the binary representations of the indices.
sponding oscillators are ES and the TO between transmitter and receiver can be described over 2

From these studies as it targets a continuous-valued computa-

ional resources as in [156], [157], [158], [159], but it differs

probabilities. By combining the results over realizations, the

al is correlated with each sequence in the set to estimate the

local gradient vector over a scaled cross polytope constructed

over standard basis vectors). At the receiver, the received sig-

Fig. 4. Synchronization imperfections in time and frequency.

In [177], the authors propose to encode a local gradient vec-

tor of length L into a sequence in an orthogonal sequence set

of size 2L for OAC, where the encoding relies on stochastic quantization. In this scheme, the transmitted sequence is chosen with the probability that is proportional to the magnitude of the elements of the normalized local gradient vector (i.e., the coefficients derived from the decomposition of the normalized local gradient vector over a scaled cross polytope constructed over standard basis vectors). At the receiver, the received signal is correlated with each sequence in the set to estimate the probabilities. By combining the results over realizations, the superposed gradient vector is obtained. It is worth noting that this scheme also separates the sign information to the orthogonal resources as in [156], [157], [158], [159], but it differs from these studies as it targets a continuous-valued computation through a probabilistic choice of the activated resource over 2L resources. In [192], the same idea is investigated for the consensus problem.

IV. WHAT ARE THE ENABLING MECHANISMS FOR OAC?

In this section, we discuss the underlying mechanisms which maintain reliable computation and elaborate on security issues.

A. Synchronization

To elaborate the effect of synchronization impairments on OAC, let f_{ES} and f_{ED,k} denote the carrier frequencies at the ES and the kth ED, respectively, where the phase of the corresponding oscillators are θ_{ES} and θ_{ED,k}, as shown in Fig. 4(a). Thus, the carrier frequency offset (CFO) and the phase offset (PO) between the kth ED and the ES are Δf_k = f_{ED,k} - f_{ES} and Δθ_k = θ_{ED,k} - θ_{ES}, respectively. Similarly, the time offset (TO) between transmitter and receiver can be described as the following: Let t_0 denote the ideal synchronization point at the ES. Assume that the synchronization point at the ES and the time-of-arrival instant of the kth ED’s signal at the ES location deviate by Δt_{ES} seconds and Δt_{ED,k} seconds, respectively. Thus, the overall TO can be expressed as Δt_k = Δt_{ED,k} - Δt_{ES} - t_0.

Let s_k(t) ∈ C be the baseband signal for the kth ED to be transmitted. Hence, the passband signal of the kth ED from the perspective of the ES can be expressed as

\[ s'_k(t) = e^{j2πf_{ES}(t - Δt_k)}e^{j2π(Δθ_k)}e^{jΔt_k}. \]  

Assume that the impulse response of the multi-path channel in the passband is given by

\[ b'_k(τ) = \sum_{p=1}^{P} a_{k,p}δ(τ - τ_{k,p}), \]  

where P is the number of paths, a_{k,p} ∈ R and τ_{k,p} ∈ R are the pth path gain and path delay for the kth ED, respectively [193, Ch. 2]. The received passband signal for the kth ED can then be expressed as

\[ r'_k(τ) = b'_k(τ) * s'_k(t) = \sum_{p=1}^{P} a_{k,p}δ(τ - τ_{k,p})e^{j2πf_{ES}τ_{k,p}}. \]  

Hence, based on (41), we can infer the following:

- The timing errors at the EDs or ES not only translate the signal in time but also cause an additional phase rotation.
- The CFO causes phase error accumulation that grows over time.
- The CFO results in an additional phase rotation, depending on the time offset and path delays.

Now, assume that the OAC is based on OFDM. For τ ∈ [0, T_{sym}], we can express the baseband signal for the kth ED as s_k(t) = \sum_{l=0}^{M-1} x_{k,l} e^{j2π l\frac{τ}{T_{sym}}}, where the T_{sym} is the symbol duration, M is the number of active subcarriers, and x_{k,l} are the transmitted symbols as given in (7). Assume that the channel is ideal, i.e., P = 1, a_{k,p} = 1, and τ_{k,p} = 0, ∀k. Also, assume that the ideal synchronization point where the N-point DFT is started to be applied to the received baseband signal is within the CP duration, i.e., at t_0 ≤ 0, as illustrated in Fig. 4(b). Under the synchronization impairments, the received symbol on the kth subcarrier at the ES can be expressed in (42), as shown at the bottom of the next page, where η_{FO,k} ≜ Δf_k T_{sym} and η_{TO,k} ≜ Δt_k / T_{sym} are the normalized CFO and TO for the kth ED, respectively, and D_N(x) ≜ \frac{1}{N} \sum_{n=0}^{N-1} e^{j2π nx/N} \sin(πx/N) is the Dirichlet sinc kernel. As can be seen from (42), the existence of CFO causes inter-carrier interference while TO due to the imperfect time-of-arrivals or the synchronization errors at the ES results in phase rotations scaled with the subcarrier index.

5In practice, an OFDM receiver intentionally backs off some duration for the DFT calculation to avoid samples from the following OFDM symbol.
On the other hand, the residual PO leads to a distortion independent from the subcarrier index. Similar observations on CFO, TO, and PO are also made in [184, Sec. VI], [194], and [153]. Since the PO, TO, and CFO affect the received signal jointly, the key OAC-related metrics such as computation rate or MSE for the computation discussed in Section III-A are also affected. However, how these metrics are affected under such offsets is not well-assessed in the literature. It is also worth emphasizing that the sensitivity of the computation to the residual offsets depends on the scheme and the function desired to be computed. For instance, for an OAC scheme that requires phase synchronization among the EDs, the PO, TO, and CFO need to be compensated very accurately, particularly for analog aggregation. Hence, this scheme would require a precise sample-level synchronization and the corresponding MSE would be sensitive to synchronization errors (See Section III-B4a). On the other hand, if the target function is an MV computation and the OAC scheme relies on keying approaches along with a non-coherent detection, it is demonstrated that maintaining the synchronization within the CP range can be sufficient and the MSE does not increase with the synchronization errors [163] (See also Section III-B4a).

In practice, it is challenging to mitigate the impacts of random TOs, CFOs, and POs on an OFDM-based OAC. However, in the case that the offsets change slowly, they may be mitigated through control loops. For example, in [194], the residual TO and PO are estimated by tracking the phase in the frequency domain before the OAC takes place. A protocol that feeds back the estimates of TO and PO to the EDs is proposed, where the ED can compensate for the residual errors. It is also worth noting that the residual CFO estimation errors cause the phase error accumulation for the packets with many OFDM symbols. To mitigate the error accumulation, extra signaling or better high-precision clocks can be utilized [153].

In the literature, the synchronization for OAC is also investigated for waveforms different from OFDM. For instance, in [184], [195], [196], the analyses are performed for a single-carrier (SC) waveform constructed with a rectangular pulse shape. In [184] and [195], to estimate the sum of the parameters under time-synchronization errors, a post-processing called whitened matched filtering and sampling is proposed assuming that the ES knows the TO for each ED. In [196], the same setup is investigated from the perspective of Bayesian approaches. In [91], sinc kernel is considered for waveform and it is proposed to recover the summation of the parameters by solving a convex semi-definite programming without any prior information on the misalignment. Note that, in [91], [184], [195], [196], the channel is not frequency selective and overall phase rotation is assumed to be compensated via channel inversions at the EDs and decoupled from the TO.

The non-coherent OAC approaches discussed in Section III-B4 provide robustness against PO as these methods do not convey the information in the phase. In [157] and [158], in order to mitigate the interference between the adjacent symbols due to the random TOs, a guard time between adjacent PPM or CSK symbols (based on DFT-s-OFDM) is proposed, respectively. Also, a larger energy calculation window than the corresponding bins at the transmitter is used for the energy calculations to accommodate the jitter, respectively. We also refer the reader to an excellent survey in [197] on the protocols regarding synchronous transmissions for low-power wireless networks, which can benefit to the implementation of OAC methods in such networks.

### B. Power Management

In this section, we investigate power management under two categories. In the first category, we discuss power management

\[ y_{\ell} = \frac{1}{N} \sum_{n=0}^{N-1} \sum_{k=1}^{M} x_{k}(t) e^{-j2\pi \frac{n}{N} \ell} \]

\[ = \frac{1}{N} \sum_{n=0}^{N-1} \sum_{k=1}^{M} e^{-j(2\pi f_{ES} \Delta t_{k} - \Delta t_{k})} e^{-j2\pi \Delta f_{k} \Delta t_{k}} e^{j2\pi \Delta f_{k} t} \]

\[ = \frac{1}{N} \sum_{n=0}^{N-1} \sum_{k=1}^{M} e^{-j(2\pi f_{ES} \Delta t_{k} - \Delta t_{k})} e^{-j2\pi \Delta f_{k} \Delta t_{k}} \sum_{l=0}^{M-1} x_{k,l} e^{j2\pi \frac{l}{T_{sym}}(t-\Delta t_{k})} e^{j2\pi \Delta f_{k} t} \]

\[ = \sum_{k=1}^{K} x_{k,\ell} e^{-j(2\pi (\eta_{FO,k}+\ell)\eta_{TO,k} + 2\pi f_{ES} \Delta t_{k} - \Delta t_{k})} D_{N}(\eta_{FO,k}) \]

\[ + \sum_{k=1}^{K} \sum_{l=0}^{M-1} x_{k,l} e^{-j(2\pi (\eta_{FO,k}+\ell)\eta_{TO,k} + 2\pi f_{ES} \Delta t_{k} - \Delta t_{k})} D_{N}(\eta_{FO,k} + l - \ell). \]
from the perspective of the receiver, e.g., power alignment at ES via power control. In the second category, we discuss power management from the perspective of transmitters and focus on the issues related to peak-to-average power ratio (PAPR) and output-power back-off (OBO).

1) Receiver Side: For power control, a perfect amplitude alignment at the ES to ensure fairness or an accurate computation, or a policy that minimizes an application-specific metric are two main objectives one can consider. In the former case, the link with the worst channel condition may dominate the performance of computation [98]. On the other hand, the latter requires a well-defined metric to be taken into account and the power control strategy becomes a function of the application. In this case, the policy can be quite diverse. For example, in [98], it is shown that the power alignment for gradient aggregation is not always necessary for the convergence of FEEL. As another example, in [83], the authors consider an online power control mechanism with the considerations of saddle regions for distributed principal component analysis (PCA) computation. In [198] and [199], for static fading channels, channel inversion that minimizes MSE for multiple transmissions is studied. It is shown that the optimal channel inversion coefficient is a function of the number of re-transmission. In [200], channel inversion that particularly minimizes the bias for a given number of re-transmissions is investigated for fast-fading channels (i.e., time diversity). In [201], it is shown that the optimal power allocation policy across OFDM subcarriers for channel inversion yields a proportional fairness scheme. In another application, Byzantine attacks are taken into account for power control [202]. We refer the reader to [24] and references therein for a comprehensive analysis on various power control methods for FEEL.

Most of the state-of-the-art power control strategies for OAC rely on a small-scale channel model when a precoder such as TCI is employed at the EDs (e.g., see [83], [202], [203], [204]). On the other hand, power control is also related to large-scale models and interactions between adjacent cells. In [205], inter-cell interference in a scenario where OAC occurs at different cells concurrently is taken into account and the optimal policies for controlling devices’ transmit power are investigated to minimize the MSE of computed function. In [206], assuming that the inter-cell interference is harmful to the OAC, the beamforming vectors and transmit powers are optimized with the consideration of proportional fairness at different cells. In [207], inter-cell interference is investigated in both UL and DL directions for a channel-inversion-based OAC. In the literature, it is also shown that inter-cell interference can actually be harnessed for OAC, as discussed in Section IV-C.

When there is a large number of devices participating in the computation, the dynamic range of a superposed signal at the ES can exceed the dynamic range of the receiver [164], [184], [208]. Reducing the transmit powers of the EDs or designing adaptive gain control for OAC are two potential solutions to address this issue. In [164], a randomization approach is proposed to spread the energy. In [208], the MSE minimization of OAC under a sum-power constraint is studied, where one of the motivations is to reduce the potential interference due to the superposed signals. In [139], the sum-power constraint is introduced for energy saving.

2) Transmitter Side: The dynamic range of the transmitted signals is directly related to the power management in a network. A transmitted signal with a large PAPR can cause a reduced cell size due to the power back-off, a higher adjacent channel interference due to saturation, and a reduced battery life to accommodate the instantaneous power fluctuations. In the literature, a few OAC schemes is analyzed from the perspective of PAPR. In [156], the randomization symbols are used to reduce the PAPR for FSK-based MV (FSK-MV) computation. It is demonstrated that if the parameters are highly correlated in the frequency domain (e.g., the elements of a stochastic gradient vector at one iteration can be highly-correlated due to the over-parameterized neural networks [209]), modulating the subcarriers with the parameters without any precaution can cause OFDM symbols with large PAPRs. In [120], [157] and [158], the properties of SC waveform and chirps are exploited to mitigate PAPR for OAC. In [158], a basic model is proposed to relate OBO to the cell size under power control: Let $P_{\text{ref}}$ and $OBO_{\text{ref}}$ be the average transmit signal power in Watts and the OBO in dB, when the link distance between the ES and an ED is $r_{\text{ref}}$ meters, respectively. Also, let $OBO_{\min}$ be the minimum OBO that does not violate ACLR or the spectrum-emission mask requirements. Based on the simplified path loss model, the received signal for the $k$th ED can be expressed as

$$P_k = \begin{cases} \left( \frac{r_k}{r_{\text{ref}}} \right)^{-\alpha+\beta} P_{\text{ref}}, & r_{\text{ref}} \leq r_k < r_{\max} \\ \left( \frac{r_{\max}}{r_{\text{ref}}} \right)^{-\alpha+\beta} P_{\text{ref}}, & r_k \geq r_{\max} \end{cases}$$

where $r_k$ is the link distance between $k$th ED and ES, $\alpha$ is the path loss exponent, $\beta \in [0, \alpha]$ is a coefficient that determines how much path loss is compensated via power control, and $r_{\max}$ is the maximum link distance beyond which the ED is unable to increase the average transmit power and it can be expressed as

$$r_{\max} = r_{\text{ref}} \times 10^{\frac{OBO_{\text{ref}} - OBO_{\min}}{10\beta}}.$$  

The parameter $r_{\max}$ determines the region in which the average received signal powers of the EDs located in this area can be aligned at the ES location. Based on (44), a smaller $OBO_{\min}$ results in a larger $r_{\max}$. However, to decrease $OBO_{\min}$, a more linear PA or an OAC scheme with low instantaneous power fluctuations is needed. In [158], based on the Rapp model, $OBO_{\min}$ values are obtained for OAC with TCI and CSK-based OAC and the trade-off between computation rate and cell size is emphasized. It is shown that the reduced-cell size due to the power back-off can deteriorate the performance of the computation due to the weak signals from the EDs with large link distances.

C. Architecture

In the literature, a majority of the papers on OAC consider either a single-cell scenario or a network where there is...
Fig. 5. Several architectures for OAC.

(a) Single cell OAC.

(b) Uplink OAC in a multi-cell environment where the interference due to the adjacent cells are harmful to the computation at the cells.

(c) Uplink OAC in a multi-cell environment where the interference due to the adjacent cells are useful to the computation at the cells.

(d) Downlink OAC for improved OAC at the cell-edge EDs.

(e) Two-tier hierarchical OAC by exploiting non-terrestrial networks.

(f) OAC in an ad-hoc network with half-duplex and full-duplex transceivers, where the network topology varies in time.

Fig. 5. Several architectures for OAC.

no cooperation among the cells. On the other hand, harnessing the interference among the cells with some coordination can also result in a more global computation while addressing large-scale fading of wireless channels. In Fig. 5, we illustrate several architectures for OAC. As compared to the single-cell scenario in Fig. 5(a), the interference due to the adjacent cells in a multi-cell scenario may be considered as harmful to the OAC as in Fig. 5(b) or harnessed for the computation as shown in Fig. 5(c). Similar to UL OAC, OAC can also be realized in DL as depicted in Fig. 5(d). In Fig. 5(e), we illustrate a two-tier hierarchical computation, where the intermediate ESs can be base stations while the final fusion node can be a satellite in a non-terrestrial network (see [210] for an example with a low-earth orbit satellite without hierarchical computation). Finally, Fig. 5(f) illustrates OAC in an ad-hoc network, e.g., to achieve consensus on the output of a function among the EDs, where the network topology can change over time and the EDs may perform OAC while transmitting to the other nodes with the consideration of full-duplex transceivers.

In [148], [149], hierarchical FEEL is investigated over multiple clusters to address the adverse effects of path loss on aggregation. In this approach, OAC is employed in each cluster, where the intermediate servers are connected to a global server to aggregate their results. In [211], hierarchical FEEL under data heterogeneity is addressed via a dynamic weighting approach. The scheme used in this paper is based on TCI. In [148], [149], [211], the impact of the inter-cluster interference is not taken into account. In [203], the inter-cluster interference is also exploited with OAC for hierarchical FEEL. In [212], it is proposed to use multiple clusters where the ESs in clusters seek a consensus for FEEL update with OAC, in addition to the OAC within the clusters among the EDs. In [69], to improve the robustness against signal attenuation for long distances, a multi-hop network is considered. In this strategy, computation occurs locally in the sensors through multiple hops till final aggregation occurs in a fusion center by exploiting the fact that a target function can be decomposed into locally computable functions.

In [2], the universality property, based on Theorem 5, is exploited to compute multiple functions in multiple sensor clusters, where each cluster is assigned to compute one of the target functions. In [159], interference in a multi-cell network is exploited for computation. In this approach, OAC is used in both UL and DL by using FSK-MV calculation. It is shown that such an approach can improve the minimization of independent objective functions and leads to better learning performance in a large area. In [213], similarly, multiple ESs are considered. OAC with TCI is considered and the analysis is performed under harmful interference due to the downlink signals for ESs. In [207], a cooperative multi-cell optimization framework is introduced to improve the average learning performance as compared to non-cooperative baseline schemes. In [125], a topology where all devices are connected with each other is investigated. The OAC relies on the existence of a large number of antennas and full-duplex capability. In [214], authors consider a topology where the EDs continue aggregation based on neighboring EDs when the EDs to ES connection is intermittently available. Although the OAC is mentioned in this study, the topology is not analyzed along with OAC. In [215], similarly, a full-duplex capability is assumed and various network topologies are investigated. In [216], OAC is applied to graph neural networks where each node in the graph locally processes the information. The OAC based on PC is investigated to improve communication efficiency and privacy preservation. While [217] uses a graph neural network with OAC to address the power allocation problem in ad-hoc networks, [218] uses the channel coefficients between the nodes as part of the graph convolution operator. In [175], [176], a consensus protocol is proposed to achieve max-consensus in a clustered network. OAC is utilized across the clusters with the consideration of full-duplex [175] and half-duplex communications [176]. For general time-varying network topology, we refer the reader to [219] and the reference therein.
D. Channel Estimation

To achieve an accurate aggregation over fading channels, accurate and fresh CSI may need to be available at the EDs and/or the ES, depending on the OAC method. While an inaccurate estimate of CSI can cause an incoherent aggregation, the aging of the CSI estimate due to the residual CFO or mobility can result in a larger overhead and limit the number of functions to be computed in one single packet.

For the OAC methods relying on precoding techniques, each ED needs its own UL CSI which can be acquired through a DL signal with a time-domain duplexing system or transmitted back to the ED based on the estimates at the ES. While the former requires a calibration procedure, the latter causes an overhead that is scaled with the number of EDs and increased latency. The inaccuracy due to either of these methods may be modeled based on some error on the ground-truth CSI. Under this model, in [220], the authors derive an optimal transceiver design that minimizes MSE with the consideration of multiple antennas at the ES with the consideration of the imperfect CSI. In [204], a strategy that takes the imperfect CSI at the EDs into account to determine the number of local update steps is proposed. In [221], the imperfect CSI is taken into account for joint device selection and transceiver design, where the main goal is to maximize the number of participating EDs under an MSE constraint. In [139], the imperfect CSI is evaluated along with RIS. In [118], an adaptive pilot re-transmission policy that offers a trade-off between wireless resources and gain in the computation accuracy is proposed.

Some OAC methods use sum-channel estimate, rather than the UL CSI for each link. For instance, in [116], it is proposed to use a procedure that optimizes the beamforming vectors at the EDs and ES iteratively by exploiting the sum-channel CSI acquisition. In this method, the ES first transmits a set of reference symbols and broadcasts its current beamforming vector. After each ED estimates the DL channel and designs its own beamforming vector, all the EDs transmit a set of common pilot symbols concurrently so that the ES can estimate the sum channel. The key observation is that the ES can update its beamforming vector based on the sum channel. Common pilots are also employed in [147], [222] to estimate the sum channel for the methods relying on channel hardening or random orthogonalization over a large number of multiple antennas. In [222], the sum channel is also used as a precoder in the DL. Instead of using pilots in DL, an echo protocol where the ES broadcast the received symbols for the sum-channel estimation back to the EDs is proposed. In the UL, the EDs design their channel inversion coefficients based on the received symbols in the DL.

E. Security

OAC relies on the superposition of the transmitted signals from the EDs. As discussed in [20], this fact has both positive and negative consequences as far as security is concerned. On one hand, the superposition in OAC promotes user privacy as the transmitted signals cannot be directly observed. On the other hand, it opens up potential adversaries to harm the computation, particularly through Byzantine attacks. This is because Byzantine attacks are launched by the nodes that are already part of the network. For example, for FEEL, if an ED’s local data are deliberately labeled incorrect (i.e., one of the data poisoning attacks) or the sign of the gradients are flipped (i.e., one of the model poisoning attacks) by an adversary or due to the failure of a node, the whole learning process can be unreliable. This problem has been studied in the FL literature by identifying the Byzantine nodes or detecting anomalies in the local signals (see [223] and [202] and the references therein for further discussions on various attacks and defense mechanisms). However, if the aggregation is handled through an OAC scheme for the same scenario, well-known defense strategies that rely on the observation of local information cannot be utilized directly as the ES only observes the superposed signal, not the signals transmitted from different EDs.

1) Byzantine Attacks: One way of achieving resiliency against Byzantine attacks relies on the geometric median, rather than the arithmetic mean, which can be expressed as

\[ z^* = \arg \min_z \sum_{k=1}^{K} \alpha_k \| z - p_k \|_2, \]  \hspace{1cm} (45)

where \( \alpha_k \) is the weight factor for the \( k \)-th parameter. It is well known that (45) can be solved with the Weiszfeld algorithm. In [62] and [81], a modified Weiszfeld algorithm is proposed for achieving a smoothed geometric median aggregation against Byzantine attacks with OAC. To calculate the geometric median in (45), the following iteration algorithm is proposed to realize with OAC:

\[ z^{(n+1)} = \frac{\sum_{k=1}^{K} \beta_k p_k}{\sum_{k=1}^{K} \beta_k}, \]  \hspace{1cm} (46)

where \( \beta_k \) is defined as

\[ \beta_k = \frac{\alpha_k}{\| \max(\| v, z^{(n)} - p_k \|_2) \|_2}, \]  \hspace{1cm} (47)

and \( v \) is a smoothing factor to prevent the denominator in (46) from yielding an unrealizable \( z^{(n+1)} \) value. The proposed scheme considers FEEL based on model aggregation, where the investigated OAC scheme is BAA with TCI (see Section III-B1a). In this approach, the \( k \)-th ED transmits the scaled model parameters \( \beta_k p_k \) and the scalar \( \beta_k \). The ES calculates (46) with OAC by using the estimates of the numerator and denominator parts and broadcast the vector \( z^{(n+1)} \). This procedure continues until a certain convergence is achieved. The proposed scheme has two main disadvantages: First, it can cause an additional delay as the algorithm should run exclusively for each communication round of FEEL. Second, it assumes that Byzantine users follow the proposed algorithm, which may not be the case in practice.

In [80], it is assumed that the ES has a reference data set and uses its own gradient as a reference vector to provide robustness against Byzantine ED. In this approach, the network divides the EDs into multiple groups in orthogonal resources and compares the distances between its own gradient and the received estimate of each group with OAC scheme that relies on analog aggregation and TCI. In this study, OAC based on BAA with channel inversion is used to reduce per-round
communication latency for each group, rather than directly addressing Byzantine attacks as done in [62] and [81].

In [202], the EDs transmit not only their standardized gradients (i.e., the variance and the mean of the local gradients are always set to 1 and 0, respectively), but also the variance that is used for the standardization. It is proposed to calculate the global gradient with a channel-inversion-based OAC while transmitting the variance information through orthogonal channels. Assuming that Byzantine attackers follow the standardization to avoid exposing themselves during the standardization stage, they would send the true mean and variance of their local gradients. Under this scenario, it is shown that TCI-based OAC for stochastic gradient descent (SGD) has limited defensive capability against Byzantine attacks as the TCI aligns the amplitude levels at the ES. The best-effort approach, i.e., using maximum power, is proposed against Byzantine attacks. The main shortcoming of the proposed approach is that a Byzantine attacker can still transmit non-standardized gradients while transmitting valid variance information.

2) Privacy: Differential privacy is a well-established metric that measures the privacy of local data sets with respect to disclosed aggregate statistics [224]. A typical approach is to randomize the disclosed statistics by adding random noise, which causes a trade-off between accuracy and privacy. For OAC, random perturbations are added to the local model parameters or gradients before transmission for FEEL [95], [225], [226]. In [95], it is shown that the privacy leakage per user scales as $O(1/\sqrt{K})$, compared to the orthogonal schemes. In [227], privacy is investigated with the consideration of random client participation and power misalignment. In [228, Lemma 3], it is shown that such an approach guarantees differential privacy and can be obtained without affecting the learning performance as long as the privacy constraint level is below a threshold. The authors also emphasize that the channel inversion for OAC under fading is beneficial for privacy. In [84], the privacy concern is addressed by incorporating channel perturbations into the optimization problem and introducing a framework that does not explicitly transmit the Hessian or the gradient to the ES. In [216], privacy-preserving signaling and privacy-guaranteed training algorithm along with OAC are investigated when a neural network is distributed across multiple nodes based on a graph. In [229], the trade-off between data privacy and training accuracy via power control optimization is investigated for channel-inversion-based OAC. In [230], differential privacy is evaluated when the devices with better channel conditions are scheduled for OAC during the training period of FEEL. In [231], it is proposed to use low-resolution analog-to-digital converters (ADCs) at the ES and digital-to-analog converter (DAC) at the EDs along with OAC for promoting privacy further.

3) Eavesdropping: Eavesdropping is one of the potential issues of OAC as an eavesdropper can overhear the computation in the wireless channel. To address this issue, in [232], data confidentiality is investigated for TBMA. The key idea in this work is that the sensors that have weaker channel gains can be utilized to confuse an eavesdropper by exploiting the independence between the desired and eavesdropping channels. A similar idea where a group of EDs with weaker channel conditions are selected as jammers is utilized for FEEL in [233]. In [234], the pre-processed symbols transmitted from the EDs, i.e., $\{p_k[n]\}, \forall k$, are intentionally distorted with jamming symbols such that the distortion can cause a substantial MSE degradation at the eavesdropper as compared to the one at the legitimate receiver, i.e., ES. The basic assumption exploited in this work is that the ES has either precise knowledge of the jamming symbols while the eavesdropper only has knowledge about the distribution of the jamming signal. Hence, under this assumption, the ES can cancel the jamming symbols or be affected by less interference as compared to the eavesdropper. The proposed scheme is investigated for computing an arithmetic average over an AWGN channel. In [235], it is proposed to use a full-duplex transceiver at the ES so that a jamming noise is transmitted to degrade the eavesdropper’s links. In [236], ES calculates an artificial noise vector such that it is projected into the null space of the channel vector after the simultaneous transmissions. The trade-off between the computation at the ES and the security against the eavesdropper is emphasized.

4) Jamming: In [237], it is proposed to use a common spreading code assigned by the ES to facilitate protected model aggregation. For this scenario, it is assumed that the adversarial user does not know the spreading code. Hence, the interference due to the adversary is suppressed in the despreading/decoding process at the ES.

V. WHAT ARE THE APPLICATIONS OF OAC?

In this section, we discuss several applications of OAC in various fields, as illustrated in Fig. 6. We also discuss the state-of-the-art demonstrations of OAC for certain applications.

A. Distributed Localization

Consider a scenario where many sensors are deployed in an area to identify the location of a radio source emitting a signal with a known transmit power via the received signal strength information (RSSI). OAC can provide a localization solution based on voting over OFDM as follows [162]:

- **Step 0**: The area is divided into a grid and all sensors know their positions and the grid structure.
- **Step 1**: Each sensor estimates the link distance between its location and source based on RSSI.
- **Step 2**: Each sensor marks the squares that intersect with the circle with the radius of the estimated link distance.
- **Step 3**: Each sensor activates the corresponding OFDM subcarriers that represent the marked squares (i.e., vote).
- **Step 4**: All the sensors transmit simultaneously.
- **Step 5**: The ES determines the location of the source by detecting the subcarrier that has the largest magnitude (i.e., MV).

The procedure above is an extension of a voting-based localization [238] with the consideration of OAC. With OAC, instead of using orthogonal channels to acquire the sensor information, the ES receives the superposed signal and obtains the locations of the radio without any extra computation. An example is illustrated in Fig. 6(a).
B. Wireless Control Systems

In control theory, a dynamic plant refers to a state-space model where the current states evolve in time. It can be modeled as a set of first-order linear difference equations, e.g., \( x(t+1) = Ax(t) + Bu(t) + w(t) \), where \( A \) and \( B \) are real-valued matrices, \( u(t) \) is plant control action, \( x(t) \) is the plant state, and \( w(t) \) is the plant noise \([12]\). A simple example of a dynamic plant is a pendulum, where the vector \( x(t) \) represents the angular position and speed of the pendulum \([239]\). The vector \( x(t) \) may involve a large number of spatially-distributed state variables. For example, for a chemical plant, the state variables may be the temperature, humidity, and pressure, while they may be atmospheric pressure, thrust, drag, speed, and acceleration for an aircraft. In \([12]\), a scenario where the stability of a dynamic plant is monitored by distributed sensors is considered. The main goal is to stabilize a potentially unstable plant over limited wireless resources by acquiring the current state as quick as possible. By incorporating the multiple-access channel into the expression of the corresponding state-space equation of the dynamic plant, OAC is exploited to support a large number of sensors. We also refer reader to \([190], [239], [240]\) for further details to dynamic plants, where the main goal is recover the state vector \( x(t) \) by using OAC. In \([241]\), OAC is applied to a general control system. The proposed scheme is concerned about the transmit power of sensors to minimize the effect of the wireless channel to the control system under a transmit power constraint. In \([101]\), distributed consensus via OAC is applied to vehicle platooning control. OAC is utilized to calculate the average positions of the vehicles, needed for the calculation of the accelerator at each vehicular to stabilize the platoon. For OAC, the authors propose to multiply the parameter (a scaled version of the position of the vehicle) with the sign of the real part of the channel at the transmitters so that a coherent addition is obtained at the receiver. Only real part of the symbols are used for computation. In \([218]\), an implementation of a graph neural network is investigated with OAC for multi-robot flocking.

C. Wireless Sensor Networks

In \([37]\), OAC is used for estimating the portion of inactive sensors in a network. For this application, all active sensors transmit the symbol 1 and the receiver estimates the number of active sensors from the received signal. If the ES knows the number of sensors in the network, it immediately estimates what portion of sensors are inactive. An example of OAC for computing the arithmetic mean of temperatures measured by 250 sensors for environment monitoring can be found in \([37]\).
In [63], several other applications of OAC like counting the number of sensors whose readings satisfy a certain threshold, a variance of the measured temperatures, or the best linear fit to the observed measurements, i.e., regression, are given in the area of WSNs. With the motivation of environment sensing and radio map construction, product-of-experts-based Gaussian process regression over a distributed network is investigated in [92], where the investigated OAC schemes are based on ZF precoder and PC. In [119], the authors consider a distributed sensing application, where all the sensors observe a linear combination of the data. In [90], [242], the mobility of unmanned aerial vehicles (UAVs) is exploited to improve the power alignment for ZF-based OAC. Cluster scheduling, association, and UAV trajectory are jointly optimized with the motivation of aligning the signals within each cluster while mitigating the inter-cluster interference. UAV-based aggregation is also studied in [96]. Similarly, UAV trajectory optimization with the consideration of multi-slot OAC is investigated in [243]. In [87], the authors consider the freshness of the aggregated data in addition to MSE in a time-varying environment for OAC-based remote monitoring applications. In [164], OAC is considered for constructing the geographical heat distribution via low-power wide-area networks with the motivation of detecting forest fires.

D. Distributed Optimization Over Wireless Networks

One of the main motivations behind OAC is the convergence of communication and computing architectures as explicitly discussed in ITU’s report for IMT-2030 [244]. The driving force for this trend is the advances in machine learning and artificial intelligence technologies such as federated learning and split learning, and their utilization over wireless networks. The major benefit gained from OAC for a distributed optimization problem is the considerable improvement in computation rate as compared to the traditional way of separating communication and computation tasks, particularly, when many EDs participates in computation.

1) Federated Learning: FL [26] is one of the most studied distributed learning frameworks. The task of model training is distributed across multiple EDs and data uploading is avoided to promote the user-privacy. Instead of data samples, EDs share a large number of local stochastic gradients or local model parameters with an ES for aggregation. For its implementation over a wireless network in general, i.e., FEEL, we refer the reader to [18], [19], [20], [21], [22], [23], [24].

For FEEL, if the communication and computation are considered as separate tasks, for each iteration, the ES needs to acquire the local model parameters (or gradients) from $K$ EDs, separately, to compute $N_f \sim 10^6 - 10^8$ functions. Hence, $N_f/K$ parameters need to be transferred in the UL and the latency grows linearly with the number of EDs for an orthogonal multiple access scheme. On the other hand, with OAC, the cost is equal to the one with a single ED as all the EDs transmit simultaneously to compute $N_f$ functions, e.g., the average of the local model parameters (or gradients). Hence, the training can be completed much faster if OAC is utilized.

One of the crucial choices for OAC to support FEEL is that the information needs to be transmitted in the UL and DL. This is because the UL information can be local gradients or local parameters, while the information broadcast to the EDs in the DL can be the updated model parameters or the aggregated gradients, leading to four different FEEL implementations. Although the gradients often have an unknown probability distribution that changes over the communication rounds [245], their magnitudes tend to decrease over iterations. Also, the gradients between adjacent communication rounds across different EDs may be highly correlated, as well as the entries of a stochastic gradient vector at one iteration (see [185], [209], [246]). In addition, even if the signs of the gradients are transmitted, convergence can be achieved [160]. These properties are exploited in the UL in several OAC papers, e.g., [78], [159], [161], [167], [168], [185], [209], [245]. In DL, the broadcasting updated model parameters can ensure that the EDs calculate the gradients based on the same model parameters. On the other hand, for multi-cell OAC, broadcasting aggregated gradients in the DL is shown to be useful for aggregation for EDs located at the cell edge [159] while promoting the personalization of the model parameters.

It is also worth noting that FEEL with OAC inherits the well-known problems in FL literature such as convergence under data and device heterogeneity, stragglers, data privacy, and various security issues. Hence, these application-specific challenges often need to be re-evaluated for a given OAC scheme. For example, it is challenging to deal with Byzantine attacks when OAC is used for FEEL since the ES does not directly observe the gradients or the model parameters. Also, training can lead to biased learning due to the imbalanced received signal powers [161], [167].

2) Split Learning: In [247], the authors investigate the idea of splitting a neural network over the EDs and ES so that the EDs can conserve the privacy of their local data sets while the computational burden is decreased on the ED side. Under a simple configuration, the EDs train the network up to a specific layer, called cut layer, and send the output of the cut layer, i.e., smashed data, with the labels to the ES. The ES completes the rest of forward step starting from a layer that concatenates and aggregates the EDs’ smashed data, i.e., aggregation layer. Afterward, the ES starts the back-propagation of the gradients from the last layer to the first layer of the ES’s neural network, and sends the gradients with respect to the smashed data to the EDs. The forward and back-propagation continue until the network converges. The main advantage of split learning (SL) over FL is that the EDs have fewer layers as compared with the ones in the FL. [248], [249], [250], [251]. Note that the model splitting also appears as vertical FL in the literature [252], where a neural network is divided and distributed across the network. User scheduling under fading channels for vertical FL is discussed in [253] without using OAC. The reader is also referred to the technical reports in [254], [255] for potential applications of SL.

SL is not heavily investigated in the state-of-the-art from the perspective of OAC. In [256], the aggregation layer of SL is proposed to be realized with BAA along with channel inversion. In this approach, to accommodate OAC, the
weighted multiplication of the aggregation layer is moved to the EDs whereas the bias addition and activation are kept at the ESs. To address the fading, the users with deep faded channels are excluded from the training. Instead of gradient or local parameter aggregation, smash data aggregation takes place for the implementation of SL over a wireless network as shown in Fig. 6(c). In [257], [258], by exploiting channel reciprocity and considering the wireless channel as part of the neural network, i.e., a form of OAC for implementing a fully-connected layer over the air, forward-backward propagation for SL is investigated over MIMO channels. The key observation in this study is that the backward propagation can still be maintained by transmitting the gradients if the channel reciprocity is maintained.

3) Other Distributed Computation Frameworks: In [216], a graph neural network where the devices correspond to the edges of a graph is investigated. Each node in the graph aggregates information from its neighboring nodes in the graph. In this study, OAC is exploited to improve the computation rate while increasing privacy. In [217], a message-passing neural network is considered to address the decentralized power allocation problem in a device-to-device network and a channel-inversion-based OAC is used. In [218], the channel coefficients are embedded into the graph convolution operator. Hence, CSI between the link becomes part of the graph neural network. In [83], OAC is utilized to perform PCA when the data is not centralized. The authors express the centralized PCA problem as a minimization problem. By using the corresponding gradients of the objective function, it is proposed to solve the minimization problem by aggregating the gradients over the air along with TCI. In [259], it is proposed to obtain a cooperative solution of a linear algebraic equation by exploiting OAC, where each agent knows only a subset of the equations. In [88], distributed primal-dual optimization is investigated along with TCI-based OAC, and applied to the energy management of a smart grid system.

In the literature, it is worth noting that there are many other statistical methods, e.g., independent component analysis, k-means, k-SVD, that can benefit from OAC. Exploration of such algorithms with the consideration of OAC is currently an open topic.

E. Wireless Data Center Networks

A data center network (DCN) manages the communications among the work nodes across data centers to store or process the files in a parallel manner. It is often implemented through a high-bandwidth wired network. On the other hand, a wired DCN has limited flexibility, cabling complexity, and device cost, which affects the scalability of DCNs. To address those issues, in [13], OAC is proposed to compute the arithmetic mean of the symbols at K source nodes for a wireless DCN.

F. Wireless Intra-Chip Computation

In [14], OAC is exploited with the motivation of addressing scaling-out wired interconnects for hyperdimensional computing. In this method, the main goal is a similarity-search task via multiple in-memory computing (IMC) cores, where the input information at each core is the bit-wise MV of the queries from different controllers (i.e., bundling) over a wireless channel as shown in Fig. 6(f). The transmitters at the controllers use BPSK symbols to transmit 0 and 1 based on their bits. The receivers at the IMC cores receive a slightly different version of the superposed symbols due to the multi-path channel. To ensure that the phases are aligned at the receivers, the authors propose to optimize the phase of transmitted symbols so that the error rate of the bit MV computation at each core is minimized.

G. Wireless Communications

1) Compute-and-Forward Relaying Scheme: With the compute-and-forward relaying strategy [1], [7], the multiple relay nodes forward the linear functions of the transmitted messages to be decoded at the destination, as illustrated in Fig. 6(d). In [260], the compute-and-forward scheme is exploited to harness the collisions for massive access. It is well-known that when a collision occurs on the channel, non-scheduling-based channel access protocols, like the carrier-sensing multiple access with collision avoidance used in Wi-Fi, require the involved devices to access the channel again using a back-off mechanism to reduce the probability of repeated collisions. To address the diminished rate in this case, in [260], collisions are exploited at the nodes and the nodes forward a linear combination of the messages to the base station along with the corresponding coefficients for decoding.

2) Physical-Layer Network Coding: PLNC is one of the well-studied applications of OAC in the area of wireless communications [5], [6], [7], [169], [170], [261]. A canonical example of PLNC is communication over a two-way relay channel. In this channel, the devices want to exchange their bits over a relay. A link-layer network-coding strategy requires three time slots to accomplish this task: In the first two time slots, the devices send their bits to the relay, sequentially. In the third slot, the relay forwards the XOR of the bits to devices. With physical-layer network coding, the same task is accomplished in two time slots by exploiting signal superposition property: The devices transmit simultaneously their signals determined based on the bits. The relay node then forwards either the superposed signal itself (i.e., analog network coding) [5] or the signal after some detection (i.e., digital network coding) [6], [7], [169], [170] to the devices. Since each of the devices knows its own signal, it can obtain the message at the other device.

In [5], the authors exploit the differential encoding along with minimum-shift keying (MSK) and discuss practical issues such as synchronization for analog network coding. In [6], it is proposed to use quadrature phase-shift keying (QPSK) symbols at the devices and the relay detects the XOR of bits, resulting in a corrupted version of the XOR of the transmitted bits. In [169], [170], FSK is utilized with the motivation of reducing strict requirements on power control, phase synchronization, and CFO. The impact of the channel on the detector design at the relay for binary FSK [169] and M-ary FSK along with an LDPC code are discussed rigorously.
In [7], Bobak shows that the nested-lattice code used in compute-and-forward relaying strategy can also be utilized in two-way relay channel to improve the reliability physical layer network coding and an excellent comparison of analog and digital network coding is provided. We also refer the reader to [16] and the references therein for the variants of physical layer network coding.

3) Overhead Reduction: In [112], the authors use OAC not only for computation but also for overhead reduction. Instead of acquiring the CSI feedback from each ED through orthogonal multiple access, they calculate the optimum receive beamforming vector at the ES by concurrent transmissions. It is shown that the feedback overhead reduction can be reduced 50 times more than the one with conventional training. In [262], the authors propose to determine the power-normalization factor for zero-forcing by calculating the minimum function through the queries discussed in Section II-B, instead of estimating the channel of each ED through orthogonal channels.

4) Cognitive Radios: In [182] and [183], a variant of Goldenbaum’s scheme discussed in Section III-B1c is used for a spectrum-sensing application for cognitive radios. In this application, the fusion center desires to detect if the primary user is active or absent by using many sensors. To this end, the symbols that are transmitted from the sensors are either the average signal power or the hard-detection activity results on the primary-user band. The information across the sensors is proposed to be aggregated over the air to achieve a time-efficient cooperative spectrum sensing. Instead of only using amplitude correction, a weighted sum based on the absolute square of the channel coefficients is incorporated to Goldenbaum’s scheme to avoid power boost due to the inversion, which effectively corresponds to maximum ratio transmission. In [262], the DFTs of the received signals at the sensors are proposed to be combined over the air for spectrum sensing. The OAC in this study relies on a zero-forcing precoder. In [92], OAC is applied to radio map construction. For this application, Gaussian process regression is considered for a scenario where the nodes are deployed in two-dimensional space and measure the RSSI of a transmitter. It is shown that the proposed approach can speed up the computation time approximately 733x than the one with separation of communication and computation tasks.

H. Security

In [60], OAC is utilized for multiplying Gaussian prime numbers to compute a secret key that can be used in any encryption process or to generate keys. As illustrated in Fig. 6(i), in this approach, each node is assigned a number and all EDs calculate the products with OAC by changing the destination node sequentially. It is argued that this can also provide physical layer security as an eavesdropper cannot directly observe the multiplication. In [263], the authors propose to use OAC for assessing the consensus in a blockchain network. With this approach, each user maps the bits in the hash to the modulation symbols in a constellation and all the users transmit simultaneously. Since the hash should be consistent among

the users, the received symbol after superposition should be one of the points in the constellation. By detection, the malicious users with inconsistent hashes are filtered out. In this study, OAC is based on zero forcing.

I. Demonstrations

In the state-of-the-art, early OAC demonstrations are mainly in the areas of WSNs and PLNC. For example, in [175], a statistical OAC is implemented with twenty-one RFID tags to compute the percentages of the activated classes that encode various temperature ranges. A trigger signal is used to achieve time synchronization across the RFIDs. In [105], Goldenbaum and Stańczak’s scheme [39] is implemented with three software-defined radios (SDRs) emulating eleven sensor nodes and a fusion center. The arithmetic and geometric means of the sensor readings are computed over a 5 MHz signal. The time synchronization across the sensor nodes is maintained based on a trigger signal and the proposed method is implemented in a field-programmable gate array. A calibration procedure is also discussed to ensure amplitude alignment at the fusion center. In [264], the summation is evaluated with a testbed that involves three SDRs as transmitters and an SDR as a receiver. The scheme used in this setup is based on channel inversion and puts limitations on the supported dynamic range to avoid exceeding the maximum transmit power. In [63], six universal software radio peripherals (USRPs) represent the sensor nodes and the experiment is repeated multiple times to emulate the effect of many sensors. The sum operation is implemented by using a binary representation of the parameters. For this experiment, AirShare protocol in [265] is utilized to ensure that all transmissions are coherent. In [262], a spectrum sensing example based on OAC is given, but the details related to the protocols for synchronization are omitted. In [266], a real-time implementation of PLNC is demonstrated. To achieve time synchronization, it is proposed to add a sufficiently long time that compensates for the transfer time between the computer and SDRs. In this method, the same clock/oscillator is connected to the co-located radios and the transmission time instants are set manually. In [267], by extending [266] into a more general framework, a time-slotted approach is proposed to maintain time-synchronization among the radios. In this method, the radios (i.e., EDs) first align their slot boundaries by compensating the time difference between the first sample of a reference packet transmitted from the relay (or ES) and the first received sample (e.g., noise) marked by the USRP hardware. By exploiting the time-stamp-based transmission feature of USRP hardware [267, Sec. III-B2], the radios transmit simultaneously at the slot boundaries. In [268], PLNC is implemented by using temperature-compensated oscillators and implementing custom blocks for accurate alignment.

To the best of our knowledge, the demonstrations of OAC schemes for FEEL are limited, but get more attention in the recent literature. In [194], a custom two-stage protocol that mitigates the TO and CFO is proposed and an OFDM-based OAC with channel inversion is investigated. In [163], a general-purpose time synchronization method
that allows a set of SDRs to transmit or receive any in-phase/quadrature data simultaneously while maintaining the baseband processing in the corresponding companion computers (CCs) is proposed. This approach relies on the detection of a synchronization waveform and passing a pre-determined number of in-phase/quadrature (IQ) samples to the CC upon its detection. All SDRs wait for a pre-determined duration for CC-based processing and transmit simultaneously the IQ samples in the SDR buffers. By implementing this synchronization method on five SDRs (i.e., Adalm Pluto) along with a control loop that mitigates TO, CFO, and power offset coarsely, the performance of FSK-MV (see Section III-B4a for details) is practically demonstrated for hand-digit recognition task based on MNIST database. The experiment shows that the test accuracy can reach more than 95% for homogeneous and heterogeneous data distributions without using channel state information at the EDs or any method for phase synchronization. The experiment is conducted in a small room, where the distances between the EDs and ES are about 5 meters, but the channel is shown to be frequency-selective. The experiment also shows that the phase rotation in the frequency domain for OFDM is a function of the time synchronization errors at the EDs and ES, which is aligned with the analysis in Section IV-A. In [82], a TCI-based OAC with two USRP N210 SDRs as EDs is used for FEEL. This setup maintains the synchronization through a cable connection between the SDRs. In [153], the aforementioned time-stamp based transmission discussed in [266], [267] is adopted to demonstrate the multi-user detection-based computation using a higher complexity receiver.

For a consensus application, in [263], srsRAN is considered for implementing a channel-inversion-based OAC with seven radios. By considering seven users, the authors propose to use user attaching procedures for time synchronization and Long-Term Evolution (LTE) frame structure. However, the details related to the time synchronization among the radios are not provided explicitly.

VI. CONCLUSION

In this section, we summarize the main takeaways based on the discussions in the previous sections and highlight several research directions.

A. Takeaways

OAC is a concept that fundamentally disrupts the traditional way of performing communication and computation as separate tasks. The main goal of OAC is to compute a multivariate function in the wireless channel and the arguments of the function are not intended to be obtained at the ES. While function computation requires two sequential steps (i.e., communication and computation) with NOMA or OMA, such separation does not occur for OAC. Hence, the main benefit gained from OAC is the improvement of the computation rate, which otherwise scales down with the number of EDs participating in the computation.

With OAC, a function that structurally matches the underlying operation that multiple access channel naturally performs can be computed. Due to the signal superposition property of wireless channels, the functions that can be computed are in the space of nomographic functions. By manipulating pre- and post-processing functions of a nomographic function, commonly-used functions such as arithmetic mean, weighted sum, norm, MV, and histogram can be computed. Although the space of nomographic function with continuous pre- and post-processing functions is limited, Kolmogorov’s superposition shows that every continuous function can be computed via multiple nomographic functions. Also, some functions such as maximum, minimum, and median can be computed systematically over iterations or based on some approximations.

Achieving reliable OAC under practical wireless channels is a challenging task since the multi-path channel between an ED and ES distorts the transmitted symbols before the signal superposition. Hence, typical linear equalization methods cannot be directly utilized for OAC to compensate for the distortion. Hence, most OAC schemes in the state-of-the-art use CSIT along with CSIR to combat the channel. The corresponding precoders are often derived based on an MSE criterion or an application-specific metric such as training loss for FEEL under certain conditions, e.g., transmit power. To deal with the distortion due to the channel at the expense of more resource consumption, there exist also blind OAC schemes that rely on channel hardening via multiple antennas or non-coherent techniques in the state-of-the-art.

From the encoding perspective, an OAC scheme can directly use continuous-valued parameters along with an analog modulation or utilize the quantized parameters with a digital modulation. In the case of analog encoding, linear or affine transformations are shown to be effective for compression if certain properties, e.g., sparsity, are present in the parameters. For digital schemes, the family of nested-lattice codes is often considered in the literature for reliable computation since the codes in this family can be made to be linear in \( \mathbb{R} \). Nevertheless, the coding for OAC is an area that requires more research. From the encoding perspective, heavy quantization, e.g., 1-bit quantization for MV computation, is shown to be an effective solution for certain applications such as distributed learning and localization while being compatible with traditional digital communication systems.

The metrics for assessing an OAC scheme differ from the traditional communication metrics such as the data rate, bit-error rate, and block-error rate since OAC aims to compute a function. Often, the performance of an OAC scheme is measured via an MSE analysis. For a digital OAC scheme, the probability of computing a single function (or a set of functions) incorrectly can also be used as a metric since the image of the function consists of a set of discrete values. The computation rate, i.e., the number of functions calculated per real dimension, is another metric that can be used for evaluating the efficiency of an OAC scheme. One can also obtain application-specific metrics such as test accuracy and convergence rate for FEEL when it is used with an OAC scheme. Such derivative metrics are beyond the scope of our survey paper.

As a working principle, OAC relies on simultaneous receptions of EDs’ signals on the same wireless resources at the
ES and shares similar enabling mechanisms for UL-orthogonal frequency division multiple access (OFDMA) and multi-user MIMO. Reliable OAC requires underlying mechanisms such as time-frequency-phase synchronization, power management, and channel estimation or feedback mechanisms to perform well. Depending on the scheme, OAC can impose very stringent requirements. For example, if the computation relies on phase synchronization among the EDs, a sample-level time-synchronization in the network must be maintained and the phase accumulation due to the residual CFOs should be addressed. On the other hand, methods that do not rely on phase synchronization are shown to provide immunity against TO and CFO impairments.

Power management for OAC has two folds: transmitter side and receiver side. From the perspective of the transmitter, an OAC scheme should consider not only maximum transmit power limitation but also the ACLR requirements and PA efficiency. From the perspective of the receiver, the power control mechanisms need to be utilized to align the received signal powers at the ES. While aligning the average signal power can be managed via typical closed-loop power control mechanisms such as the ones in 4G LTE and 5G NR, a perfect amplitude alignment among the EDs at ES can be challenging as it requires accurate CSIT and/or low-latency feedback in time-varying fading channel conditions, and channel inversion under the transmit power limitations. Network topology for OAC has also a major impact on the computation as it extends the basic single-cell OAC to a higher-complexity computation framework that involves many fusion nodes. However, the implications of OAC for a large-scale multi-cell computation are largely unknown.

OAC has both negative and positive aspects in terms of security. On the positive side, user privacy is promoted as the transmitted signals cannot be directly observed due to the superposition. On the negative side, potential adversaries can harm the computation, particularly via Byzantine attacks. Similarly, a jammer can interfere with the superposed signals or an eavesdropper can overhear the computation in the wireless channel.

In the state-of-the-art, OAC has been considered for a wide variety of applications such as localization (e.g., voting-based distributed localization), wireless control systems (e.g., dynamic plants), wireless sensor networks (e.g., environment monitoring and UAV-trajectory optimization), distribution optimization (e.g., FEEL), wireless data center and intra-chip computation (e.g., wireless computation and similarly search), wireless communication systems (e.g., PLNC and spectrum sensing), and security (e.g., key generation). Among these applications, distributed optimization is currently the leading use case of OAC due to the advances in machine learning and artificial intelligence and the desire to use these techniques over wireless networks.

B. Research Directions

The results in the state-of-art overall advocate that OAC can address latency issues by improving the computation rate. On the other hand, OAC needs to be evaluated further along with enabling mechanisms, applications, and corresponding algorithms. To this end, three major research directions that one can pursue are summarized as follows:

Direction 1 (OAC Schemes With the Consideration of Practical Limitations): In the literature, OAC primarily is investigated theoretically under certain assumptions. Hence, some of the practical aspects may be omitted. To address this issue, the methods need to be evaluated under more challenging scenarios or designed with the considerations of imperfections and practical limitations. For instance, in practice, imperfect CSI, the residual TO, CFO, PO, mobility, and PA non-linearity may be inevitable and their impacts on the performance depend on the robustness of the OAC scheme and the corresponding applications to these imperfections. Another way of addressing this issue is to generate convincing results through demonstrations. A plausible demonstration needs not only the implementation of the OAC scheme but also the design of the underlying protocols that maintain signal superposition. Hence, the work in this area often involves developing the corresponding protocols in addition to the OAC scheme itself. Another practical challenge in this direction is that it is often not trivial to configure multiple standard SDRs for simultaneous transmissions. Even if there are some proof-of-concept OAC demonstrations, there is no widely-accepted multiple access channel testbed or platform to test different OAC schemes under realistic scenarios in controllable environments.

Direction 2 (Algorithms With the Consideration of OAC): Another area that can be improved is the algorithms in the applications. In the literature, the algorithms for many applications are not designed with the consideration of OAC. On the other hand, the algorithm can be designed to facilitate OAC and relax the constraints for enabling mechanisms. For example, distributed training by MV with signSGD, a machine learning concept, is more compatible with digital modulation as compared to the one with SGD and results in various OAC schemes. Similarly, the implementations of plain FL based on stochastic gradients (i.e., FedSGD) or parameter aggregations (FedAve) are mathematically equivalent to each other. However, the corresponding algorithms for FEEL relying on OAC can perform differently as the gradients and parameters have different statistical characteristics that can be exploited for OAC. The algorithms that facilitate multi-cell computation under various network architectures are also needed to be developed.

Direction 3 (Protocols for OAC With the Consideration of Standards): To this date, OAC has not been used in any communication standard or a commercial system. In fact, OAC has recently been discussed in AI/ML Technical Interest Group for IEEE 802.11 for distributed learning [269]. In 3GPP meetings, use cases and potential requirements for 5G to support machine learning applications under three main categories, i.e., FEEL, SL, and model distribution, are studied [254], [255], which highlights the need for a comprehensive system optimization under communication and computation constraints. Similarly, in International Telecommunication Union (ITU)’s report [244], the convergence of communication and computing architecture in International Mobile Telecommunications (IMT) systems...
towards 2030 is emphasized. Currently, it is not clear if future wireless networks will utilize OAC to facilitate this convergence or if the existing procedures can support an OAC scheme reliably or not. This is because the current wireless standard protocols in the state-of-the-art are designed by assuming that the communication is separated from the computation. Hence, further evaluations of the underlying systems and the enablers for OAC are needed. To achieve a standardized OAC, the procedures for time-frequency synchronization, power control, channel estimation, calibration, re-transmissions, compression, and security aspects along with the architectures need to be re-evaluated. A standardized OAC can insure interoperability among devices from different manufacturers for a large body of applications.

REFERENCES

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

[2] M. Goldenbaum, H. Boche, and S. Stanczak, “Harnessed interference for analog function computation in wireless sensor networks,” *IEEE Trans. Signal Process.*, vol. 61, no. 20, pp. 4893–4906, Oct. 2013.

[3] S.-W. Jeon, C.-Y. Wang, and M. Gastpar, “Computation over Gaussian networks with orthogonal components,” *IEEE Trans. Inf. Theory*, vol. 60, no. 12, pp. 7841–7861, Dec. 2014.

[4] 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.

[5] S. Katti, S. Gollakota, and D. Katabi, “Embracing wireless interference: Analog network coding,” in *Proc. Conf. Appl. Technol. Archit. Protocols Comput. Commun. (SIGCOMM)*, 2007, pp. 397–408. [Online]. Available: https://doi.org/10.1145/1282380.1282425

[6] S. Zhang, S. C. Liew, and P. P. Lam, “Hot topic: Physical-layer network coding,” in *Proc. 12th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2006, pp. 358–365. [Online]. Available: https://doi.org/10.1145/1161089.1161129

[7] 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.

[8] B. Nazer, A. G. Dimakis, and M. Gastpar, “Local interference can accelerate Gossip algorithms,” *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 876–887, Aug. 2011.

[9] G. Mergen and L. Tong, “Type based estimation over multiaccess channels,” *IEEE Trans. Signal Process.*, vol. 54, no. 2, pp. 613–626, Feb. 2006.

[10] G. Mergen, V. Navare, and L. Tong, “Asymptotic detection performance of type-based multiple access over multiaccess fading channels,” *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 1081–1092, Mar. 2007.

[11] G. Zhu, J. Xu, K. Huang, and S. Cui, “Over-the-air computing for wireless data aggregation in massive IoT,” *IEEE Wireless Commun.*, vol. 28, no. 4, pp. 56–57, Aug. 2021.

[12] S. Cui, Y. Deng, and E. Ozdemir, “The magic of superposition: A survey on simultaneous transmission based wireless systems,” *IEEE Access*, vol. 10, pp. 79760–79794, 2022.

[13] B. Yamansavasacal, A. Ozbudak, and E. Ersoy, “Air Computing: A Survey on a New Generation Computation Paradigm in 6G Wireless Networks.” [Online]. Available: https://arxiv.org/abs/2209.04640

[14] D. A. Sprechler, “A representation theorem for continuous functions of several variables,” *Proc. Amer. Math. Soc.*, vol. 16, no. 2, pp. 200–203, 1965.

[15] R. Buck, “Approximate complexity and functional representation,” *J. Math. Anal. Appl.*, vol. 70, no. 1, pp. 280–298, 1979.

[16] M. F. Frey, “Distributed function approximation over noisy channels,” Ph.D. dissertation, Dept. Comput. Sci., Technische Universität Berlin, Berlin, Germany, 2023.

[17] R. C. Buck, “Nomographic functions are nowhere dense,” *Proc. Amer. Math. Soc.*, vol. 85, no. 2, pp. 195–199, 1982.
U. Altun, S. T. Basaran, G. K. Kurt, and E. Ozdemir, “Scalable secret
key generation for wireless sensor networks,” IEEE Syst. J., vol. 11, no. 2, pp. 596–605, Dec. 2017.

Y. Sternfeld, “Dimension, superposition of functions and separation of
points, in compact metric spaces,” Proc. Amer. Math. Soc., vol. 50, no. 1, pp. 13–53, 1965.

D. Hilbert, “Mathematical problems,” Bull. Amer. Math. Soc., vol. 8, no. 10, pp. 437–479, 1902.

D. A. Sprecher, “On the structure of continuous functions of several
variables,” Trans. Amer. Math. Soc., vol. 115, pp. 340–355, Mar. 1965.

J. Braun and M. Griebel, “On a constructive proof of Kolmogorov's
superposition theorem,” Constr. Approx., vol. 30, no. 3, pp. 653–675, 2009.

J. Braun, “An application of Kolmogorov's superposition theorem to
function reconstruction in higher dimensions,” Ph.D. dissertation, Rheinische Friedrich-Wilhelms Universität Bonn, Bonn, Germany, 2009.

F. Girosi and T. Poggio, “Representation properties of networks:
Best Approximation by Linear Superpositions,” Proc. IEEE, vol. 118, no. 4, pp. 653–675, 2021.

S. Y. Khavinson, Best Approximation by Linear Superpositions
(Approximate Homography) (Translations of Mathematical Monographs). Providence, RI, USA: Amer. Math. Soc., 1996.

S. Limmer, J. Mohammadi, and S. Stanczak, “A simple algo-
gerithm for approximation by normomonic functions,” in Proc. IEEE Ann. Allerton Conf. Commun. Control Comput. (Allerton), 2015, pp. 465–469.

I. Bjelaković, M. Frey, and S. Stafieck, “Distributed approximation of
functions over fast fading channels with applications to dis-
tributed learning and the max-consensus problem,” in Proc. IEEE Ann. Allerton Conf. Commun. Control Comput. (Allerton), 2019, pp. 1146–1153.

A. Giridhar and P. R. Kumar, “Computing and communicating func-
tions over sensor networks,” IEEE J. Sel. Areas Commun., vol. 23, no. 4, pp. 755–764, Apr. 2005.

A. Giridhar and P. Kumar, “Toward a theory of in-network computation in
wireless sensor networks,” IEEE Commun. Mag., vol. 44, no. 4, pp. 98–107, Apr. 2006.

U. Altun, S. T. Basaran, G. K. Kurt, and E. Ozdemir, “Scalable secret
key generation for wireless sensor networks,” IEEE Syst. J., vol. 16, no. 4, pp. 6031–6041, Dec. 2022.

Z. Liu, Q. Lan, A. E. Kalro, P. Popovski, and K. Huang, “Over-
the-Air Multi-View Pooling for Distributed Sensing.” 2023. [Online]. Available: https://arxiv.org/abs/2302.09771.

S. Huang, Y. Zhou, T. Wang, and Y. Shi, “Byzantine-resilient fed-
erate machine learning via over-the-air computation,” in Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops), 2021, pp. 1–6.

O. Abari, H. Rahul, and D. Katabi, “Over-the-Air Function
Computation in Sensor Networks.” 2016. [Online]. Available: https://arxiv.org/abs/1612.02307.

C. Lin, N. Zhao, Y. Chen, F. R. Yu, and G. Wei, “Over-the-air
computation for IoT networks: Developing multiple functions with
antenna arrays,” IEEE Internet Things J., vol. 5, no. 6, pp. 5296–5306, Dec. 2018.

N. Agrawal, M. Frey, and S. Stafieck, “A scalable max-consensus
protocol for noisy ultra-dense networks,” in Proc. IEEE Int. Workshop
Signal Proc. Adv. Wireless Commun. (SPAWC), 2019, pp. 1–5.

S.-W. Jeon and B. C. Jung, “Adaptive analog function computation via
fading multiple-access channels,” IEEE Commun. Lett., vol. 22, no. 1, pp. 213–216, Jan. 2018.

L. Chen, X. Qin, and G. Wei, “A uniform-forcing transceiver design for
over-the-air function computation,” IEEE Wireless Commun. Lett., vol. 7, no. 6, pp. 942–945, Dec. 2018.

W. J. Liu, X. Zhang, Y. Li, and B. Vucetic, “Over-the-air computation
systems: Optimization, analysis and scaling laws,” IEEE Trans. Wireless
Commun., vol. 19, no. 8, pp. 5488–5502, Aug. 2020.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
S. Jing and C. Xiao, “Transceiver beamforming for over-the-air computation with one-sided CSI,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), 2022, pp. 1271–1276.

T. Sery and K. Cohen, “On analog gradient descent learning over multiple access fading channels,” IEEE Trans. Signal Process., vol. 68, pp. 2897–2911, 2020.

M. Seif, R. Tandon, and M. Li, “Wireless federated learning with local differential privacy,” in Proc. IEEE Int. Symp. Inf. Theory (ISIT), 2020, pp. 2604–2609.

X. Zhong, X. Yuan, H. Yang, and C. Zhong, “UV-assisted hierarchical aggregation for over-the-air federated learning,” in Proc. IEEE Global Commun. Conf., Dec. 2022, pp. 1–6.

C. Feres, B. C. Levy, and Z. Ding, “Over-the-air collaborative learning in joint decision making,” in Proc. IEEE Global Commun. Conf. (GLOBECOM), 2022, pp. 1–6.

W.-T. Chang, M. S. E. Mohamed, and R. Tandon, “On the necessity of aligning gradients for wireless federated learning,” in Proc. IEEE Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC), 2021, pp. 226–230.

R. Paul, Y. Friedman, and K. Cohen, “Accelerated gradient descent learning over multiple access fading channels,” IEEE J. Sel. Areas Commun., vol. 16, no. 2, pp. 532–547, Feb. 2022.

R. Jiang and S. Zhou, “Cluster-based cooperative digital over-the-air aggregation for wireless federated edge learning,” in Proc. IEEE/CIC Int. Conf. Commun. China (ICCC), 2020, pp. 887–892.

J. Lee, Y. Jang, H. Kim, S.-L. Kim, and S.-W. Ko, “Over-the-Air Consensus for Distributed Vehicle Platooning Control (Extended Version),” 2022. [Online]. Available: https://arxiv.org/abs/2211.06622

M. Goldenbaum and S. Stanczak, “Computing functions via SISO multiple-access channels: How much channel knowledge is needed?,” in Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP), 2010, pp. 3394–3397.

M. Goldenbaum and S. Stanczak, “On the channel estimation effort for analog computation over wireless multiple-access channels,” IEEE Wireless Commun. Lett., vol. 3, no. 3, pp. 261–264, Jun. 2014.

G. Shi, S. Guo, J. Ye, N. Saeed, and S. Dang, “Multiple parallel federated learning via over-the-air computation,” IEEE Open J. Commun. Soc., vol. 3, pp. 1325–1346, 2022.

A. K. Das and T. W. Kailath, “Precoder design for correlated data aggregation via over-the-air computing in sensor networks,” in Proc. IEEE Global Commun. Conf. Workshops (GLOBECOM WKSHPS) Workshop Wireless Commun. Distrib. Intell., Dec. 2022, pp. 1–6.

X. Liu, W. Ni, H. Tian, and Y. Wu, “Simultaneous federated learning and information transmission over time-varying MIMO channels,” in Proc. IEEE Global Commun. Conf. Workshops (GLOBECOM WKSHPS) Edge Learn. 5G Mobile Netw. Beyond, Dec. 2022, pp. 1–6.

Q. W. Liu, B. Vucetic, and Y. Li, “Over-the-air computation via broadband channels,” IEEE Wireless Commun. Lett., vol. 10, no. 10, pp. 2150–2154, Oct. 2021.

N. Zhang and Y. Malitsky, “Over-the-Air Computation With Multiple Receivers: A Space-Time Approach,” 2022. [Online]. Available: https://arxiv.org/abs/2208.11751

S. Tang, P. Popovski, C. Zhang, and S. Obana, “Multi-Slot Over-the-Air Computation in Fading Channels,” 2021. [Online]. Available: https://arxiv.org/abs/2101.03559

J. Joung and J. Fan, “Over-the-air computation strategy using space-time line code for data collection by multiple unmanned aerial vehicles,” IEEE Access, vol. 9, pp. 105230–105241, 2021.

Y. Saifullah, Y. He, A. Boag, G.-M. Yang, and F. Xu, “Recent progress in reconﬁgurable and intelligent metasurfaces: A comprehensive review of tuning mechanisms, hardware designs, and applications,” Adv. Sci., vol. 9, no. 33, 2022, Art. no. 2203747.

W.-T. Chang, M. S. E. Mohamed, and R. Tandon, “Reconfigurable intelligent surface enabled federated learning: A unified communication-learning design approach,” IEEE Trans. Wireless Commun., vol. 20, no. 11, pp. 7595–7609, Nov. 2021.

W. Liu, X. Yuan, and Y.-J. A. Zhang, “Reconfigurable intelligent surface enabled federated learning: A unified communication-learning design approach,” IEEE Trans. Wireless Commun., vol. 19, no. 12, pp. 1683–1697, Dec. 2020.

W. Liu, X. Yuan, and Y.-J. A. Zhang, “CSSI-free model aggregation for federated edge learning via reconﬁgurable intelligent surface,” IEEE Wireless Commun. Lett., vol. 10, no. 11, pp. 2440–2444, Nov. 2021.

F. Zhu, Y. Zhao, W. Xu, and X. You, “CSSI-free model aggregation for multi-RIS-assisted over-the-air computation,” in Proc. IEEE Int. Symp. Wireless Technol. Sys. (ISWΤS), 2021, pp. 1–5.

W. Zhang, J. Xu, W. Xu, X. You, and W. Fu, “Worst-case design for RIS-aided over-the-air computation with imperfect CSI,” IEEE Commun. Lett., vol. 26, no. 9, pp. 2136–2140, Sep. 2022.
K. Wei et al. “Vertical Federated Learning: Challenges, Methodologies and Experiments.” 2022. [Online]. Available: https://arxiv.org/abs/2202.04309

Z. Zhang, G. Zhu, and S. Cui. “Low-Latency Cooperative Spectrum Sensing Via Truncated Vertical Federated Learning.” 2022. [Online]. Available: https://arxiv.org/abs/2208.03694

3GPP. “TR 22.876 V0.1.0—Study on AI/ML Model Transfer-Phase 2.” Sep. 2022. [Online]. Available: https://ftp.3gpp.org/Specs/archive/22_series/22.876/22876-010.zip

3GPP. “TR 22.876 V18.2.0—Study on Traffic Characteristics and Performance Requirements for AI/ML Model Transfer in 5GS (Release 18).” Dec. 2021. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/22_series/22.876/22876-18.2.0.zip

M. Krouka, A. Elgabli, C. B. Issaid, and M. Bennis. “Communication-efficient split learning based on analog communication and over the air aggregation.” in Proc. IEEE Global Commun. Conf. (GLOBECOM), 2021, pp. 1–6.

Y. Yang, Z. Zhang, and Z. Yang. “Over-the-air split learning with MIMO-based neural network and constellation-based activation.” in Proc. IEEE Int. Workshop Mach. Learn. Signal Process., 2022, pp. 1–7.

Y. Yang et al. “Over-The-Air Split Machine Learning in Wireless MIMO Networks.” 2022. [Online]. Available: https://arxiv.org/abs/2210.04742

F. Molinari and J. Raisch. “Exploiting wireless interference for distributively solving linear equations,” IFAC PapersOnLine, vol. 53, no. 2, pp. 2999–3006, 2020.

M. Goldenbaum, P. Jung, M. Racleula-Motoc, J. Schreck, S. Staicuak, and C. Zhou. “Harnessing channel collisions for efficient massive access in 5G networks: A step forward to practical implementation,” in Proc. IEEE Int. Symp. Turbo Codes Iterative Inf. Process. (ISTC), 2016, pp. 335–339.

S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft. “XORs in the air: Practical wireless network coding,” IEEE/ACM Trans. Netw., vol. 16, no. 3, pp. 497–510, Jun. 2008.

L. Chen, N. Zhao, Y. Chen, F. R. Yu, and G. Wei, “Over-the-air computation for cooperative wideband spectrum sensing and performance analysis,” IEEE Trans. Veh. Technol., vol. 67, no. 11, pp. 10603–10614, Nov. 2018.

X. Xie, C. Hua, P. Gu, and W. Xu. “AirCon: Over-The-Air Consensus for Wireless Blockchain Networks.” 2022. [Online]. Available: https://arxiv.org/abs/2211.16700

U. Altun, S. T. Başaran, H. Alakoca, and G. K. Kurt. “A testbed based verification of joint communication and computation systems,” in Proc. IEEE Telecommun. Forum (TELFOR), 2017, pp. 1–4.

O. Abari, H. Rahul, D. Katabi, and M. Pant, “AirShare: Distributed coherent transmission made seamless,” in Proc. IEEE Conf. Comput. Commun. (INFOCOM), 2015, pp. 1742–1750.

L. Lu et al., “Real-time implementation of physical-layer network coding,” in Proc. ACM Softw. Radio Implement. Forum (SRIF), 2013, pp. 71–76.

L. You, S. C. Liew, and L. Lu, “Reliable physical-layer network coding supporting real applications,” IEEE Trans. Mobile Comput., vol. 16, no. 8, pp. 2334–2350, Aug. 2017.

Y. Tan, S. C. Liew, and T. Huang, “Mobile lattice-coded physical-layer network coding with practical channel alignment,” IEEE Trans. Mobile Comput., vol. 17, no. 8, pp. 1908–1923, Aug. 2018.

A. Şahin. “Wireless for ML—Over-The-Air Computation.” Sep. 2022. [Online]. Available: https://mentor.ieee.org/802.11/dcn/22/11-22-1483-01-aiml-wireless-for-ml-over-the-air-computation.pptx