Node Scheduling for AF-based Over-the-Air Computation

Suhua Tang, *Senior Member, IEEE,* Hiroyuki Yomo, *Member, IEEE,* Chao Zhang, and Sadao Obana

Abstract—Over-the-air computation is a promising technique for efficiently aggregating data in sensor networks. This method requires that signals from all nodes arrive at the sink aligned in signal magnitude, which faces the reliability issue, especially in times of channel fading. To solve this problem, in this paper, we propose an amplify-and-forward based relay, Coherent Relay with Node Scheduling (CohR-NS), where a relay node is used to help forward signals of multiple nodes. Relay transmission power (TP) increases with the number of nodes using the relay, which is a bottleneck. We investigate how relay TP changes with relay position, and under the constraint of relay TP, study (i) how to select nodes to use relay when not all nodes requiring a relay can be supported simultaneously, (ii) how to select more nodes to use relay so as to reduce node TP, when there is a surplus in relay TP. We formulate this as an ILP (integer linear programming) problem, propose an efficient heuristic method, and confirm its effectiveness by simulation evaluation.

Index Terms—Over-the-air computation, amplify and forward, relay, node scheduling

I. Introduction

In order to create smart cities, many sensors will be deployed to monitor the environment, meanwhile it is necessary to collect and process large amounts of data from sensors and control some actuators accordingly. This faces the scalability issue when the number of nodes increases quickly.

Over-the-air computation (AirComp) is a promising method for scalable data aggregation by integrating data collection and processing [1]. Generally, AirComp relies on the superposition property of wireless channel and only supports the sum operation. But AirComp can realize nomographic functions [2], [3] based on proper preprocessing and post-processing. In addition, it is also possible to learn unknown aggregation methods from real data by using machine learning [4]. Although digital communication is more reliable and is preferred in modern communication systems, analog amplitude modulation is usually used in AirComp [5] because it facilitates the exploitation of signal superposition in the air.

S. Tang is with Graduate School of Informatics and Engineering, The University of Electro-Communications, Japan. (e-mail: shtang@uec.ac.jp).

H. Yomo is with Faculty of Engineering Science, Kansai University, Japan.

C. Zhang is with School of Information and Communications Engineering, Xi'an Jiaotong University, China.

S. Obana is with The University of Electro-Communications, Japan.

AirComp requires that signals from all nodes arrive at the sink simultaneously and be aligned in signal magnitude. However, this is difficult when some nodes face channel fading. Transmission power (TP) control [6], [7], which pre-amplifies signals, helps to partially solve this problem, but it alone cannot well deal with deep fading.

Among the methods [8] to improving the reliability of AirComp, one is CohRelay, an amplify-and-forward (AF) based relay method [9], where one relay node is used to help multiple nodes, forwarding their signals to the sink. It was shown that the relay TP increases with the number of nodes using relay, which becomes the bottleneck of the system. But the constraint of relay TP is not explicitly considered.

In this paper, we propose a Coherent Relay with Node Scheduling (CohR-NS) method to optimize computation mean squared error (MSE) of the AF-based AirComp. We start with the optimal solution where all nodes transmit their signals directly. Then, we use the relay to help nodes whose signal magnitudes are misaligned, and further study how to schedule fewer or more nodes to use the relay, according to relay TP. To the best of our knowledge, this is the first work on node scheduling for AirComp.

The contribution of this paper is three-fold, as follows:

- We investigate how relay TP changes with relay position, which enables new policies for node scheduling.
- We formulate node scheduling as an ILP problem and propose an efficient heuristic method.
- We analyze and discuss the effectiveness, complexity, and optimality of the proposed scheduling method.

Simulation evaluations confirm that the proposed method effectively reduces MSE meanwhile suppresses node TP.

II. Relay Model for AirComp

We consider a task of data aggregation in a sensor network consisting of K nodes, a sink d and a relay r, as shown in Fig. 1. Sink d collects data from all nodes and computes a function of these data, sum as an example in this paper. Nodes near the sink directly transmit their signals to the sink, but nodes far from the sink may experience deep channel fading. A relay r is used to help these nodes, by forwarding an extra copy of their signals. In the transmission, both relay r and nodes face the constraint of maximal transmission power. Therefore, all nodes are divided into two groups. A node $k \in \mathbb{N}_d$ will only directly transmit its signal to sink d while a node $k \in \mathbb{N}_r$ will use relay r, besides the direct transmission. Here, $\mathbb{N}_r \cup \mathbb{N}_d = \{1, 2, \cdots, K\}$, and $\mathbb{N}_r \cap \mathbb{N}_d$ is empty. All nodes, relay r and sink d use a single antenna. The extension of this model to support multiple antennas at the sink is left as future work.
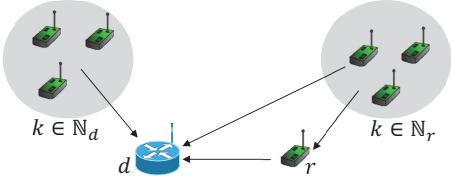
Fig. 1: Relay for AirComp in a sensor network consisting of K nodes, a sink d and a relay r.

We assume that (i) channel coefficients, $h_{k,r} \in \mathbb{C}$ ($\mathbb{C}$ is the set of complex numbers) from node k to relay r, $h_{k,d} \in \mathbb{C}$ from node k to sink d, and $h_{r,d} \in \mathbb{C}$ from relay r to sink d, are constant within the period of transmission, and (ii) sink d knows all the channel coefficients. This assumption on the availability of channel coefficients is common in previous studies on AirComp [6], [7].

The whole transmission is divided into two slots, analogy to the conventional unicast AF method [10]. In the first slot, all nodes in $\mathbb{N}_r$ simultaneously transmit their signals to relay r, and node k transmits a signal $x_k \in \mathbb{C}$, which has zero mean and unit variance ($\mathbb{E}\left\{|x_k|^2\right\} = 1, |x_k| \leq v$), using a Tx-scaling factor $b_{k,1}$. After applying a Rx-scaling factor $a_r \in \mathbb{C}$, the computation result at relay r is

$$y_r = a_r \cdot \left(\sum_{k \in \mathbb{N}_r} h_{k,r} b_{k,1} x_k + n_r \right). \qquad (1)$$

In the second slot, all nodes, including both nodes in $\mathbb{N}_r$ and those in $\mathbb{N}_d$, simultaneously transmit their signals to sink d, and node k uses a Tx-scaling factor $b_{k,2}$. Meanwhile, relay r also forwards the signals (y_r) received in the first slot, using a Tx-scaling factor b_r. Then, after applying a Rx-scaling factor $a_d \in \mathbb{C}$, the computation result at sink d is

$$y_d = a_d \cdot \left(\sum_{k \in \mathbb{N}_r \cup \mathbb{N}_d} h_{k,d} b_{k,2} x_k + h_{r,d} b_r y_r + n_d \right). \qquad (2)$$

Here, n_r and n_d are additive white Gaussian noise with zero mean and variance being σ^2. This result can be rewritten as

$$y_d = \sum_{k \in \mathbb{N}_r \cup \mathbb{N}_d} \beta_k x_k + a_d n_d + a_r' n_r, \qquad (3)$$

$$\beta_k = \begin{cases} a_d h_{k,d} b_{k,2} + a_r' h_{k,r} b_{k,1} & k \in \mathbb{N}_r \\ a_d h_{k,d} b_{k,2} & k \in \mathbb{N}_d \end{cases},$$

where $a_r' = a_d h_{r,d} b_r a_r$ is the equivalent Rx-scaling factor for the relayed signals. The computation MSE is computed as the expectation of squared difference between y_d and $\sum_k x_k$, with respect to random signals (x_k) and noises (n_d, n_r), as follows:

$$\text{MSE} = \mathbb{E}\left\{ \left| y_d - \sum_{k=1}^{K} x_k \right|^2 \right\} = \text{MSE}_r + \text{MSE}_d, \qquad (4)$$

$$\text{MSE}_r = \sum_{k \in \mathbb{N}_r} |1 - a_d h_{k,d} b_{k,2} - a_r' h_{k,r} b_{k,1}|^2 + \sigma^2 |a_r'|^2,$$

$$\text{MSE}_d = \sum_{k \in \mathbb{N}_d} |1 - a_d h_{k,d} b_{k,2}|^2 + \sigma^2 |a_d|^2,$$

where it is assumed that signals are uncorrelated and independent of noises. Because it is possible to adjust the phase of $b_{k,1}$ and $b_{k,2}$ to ensure that $a_r' h_{k,r} b_{k,1}$ and $a_d h_{k,d} b_{k,2}$ are positive real numbers, for simplicity, it is assumed that $a_d, a_r', b_{k,1}, b_{k,2}, h_{k,r}, h_{k,d}, h_{r,d}$ all belong to $\mathbb{R}^+$, the set of positive real numbers, in the analysis.

The instantaneous TP, $|b_{k,i} x_k|^2, i = 1, 2$, should be no more than P', the maximal TP. Let $P_{\max}$ denote P'/v^2. Then, we have $|b_{k,i}|^2 \leq P'/v^2 = P_{\max}$. A node $k \in \mathbb{N}_d$ only transmits in the second slot, $b_{k,1} = 0$ and $|b_{k,2}|^2 \leq P_{\max}$. A node $k \in \mathbb{N}_r$ transmits its signal twice, and the overall TP constraint, $|b_{k,1}|^2 + |b_{k,2}|^2 \leq P_{\max}$, is also applied. At the relay node, the power to transmit a signal x_k is

$$\text{TxR}_k = \mathbb{E}\{|b_r a_r h_{k,r} b_{k,1} \cdot x_k|^2\} = \left| \frac{a_r'}{a_d h_{r,d}} h_{k,r} b_{k,1} \right|^2. \qquad (5)$$

The selection of $\mathbb{N}_r$ should both ensure $\sum_{k \in \mathbb{N}_r} \text{TxR}_k \leq P_{\max}$ and minimize MSE. Then, the problem is how to find optimal parameters ($a_r', a_d, b_{k,1}, b_{k,2}$) and node scheduling ($\mathbb{N}_r$) that minimize MSE, under the power constraint, as follows:

$$\operatorname*{argmin}_{a_r', a_d, b_{k,1}, b_{k,2}, \mathbb{N}_r} \text{MSE}, \qquad (6)$$

$$\text{s.t. } |b_{k,1}|^2 + |b_{k,2}|^2 \leq P_{\max}, \forall k \in \mathbb{N}_r,$$

$$b_{k,1} = 0, |b_{k,2}|^2 \leq P_{\max}, \forall k \in \mathbb{N}_d,$$

$$\sum_{k \in \mathbb{N}_r} \text{TxR}_k \leq P_{\max}.$$

III. Scheduling Nodes to Use Relay

To solve the problem in (6), it is necessary to first decide $\mathbb{N}_r$, which is called node scheduling.

A. Initializing $\mathbb{N}_r$

Without relay, $a_r' = 0$, all nodes directly transmit their signals to the sink, and MSE is computed as follows,

$$\text{MSE} = \sum_{k=1}^{K} |1 - a h_{k,d} b_k|^2 + \sigma^2 |a|^2, \qquad (7)$$

where b_k and a are the Tx- and Rx-scaling factors, respectively. Then, the optimal Rx-scaling factor $a = a_0$ is computed by considering misalignment in signal magnitude [6], [7]. A straightforward relay method is to let nodes whose signals are misaligned form a set $\mathbb{N}_r$ and the other nodes form a set $\mathbb{N}_d$.

B. Deciding Node Transmission Power

When using a relay, node k divides its TP into two parts, $b_{k,1}$ and $b_{k,2}$, and transmits its signal twice. Sink d receives two copies of the same signal (one directly and the other via relay). With properly set parameters, the two copies are in phase and add constructively. This is called coherent relay, which helps to achieve larger signal magnitude with the same

overall power. The signal magnitude is a sum of two parts, $a'_r h_{k,r} b_{k,1}$ (relayed signal) and $a_d h_{k,d} b_{k,2}$ (direct signal), so that the overall magnitude approaches 1.0. To effectively use node TP, it is necessary to properly compute $b_{k,1}$ and $b_{k,2}$, under the constraint $|b_{k,1}|^2 + |b_{k,2}|^2 \leq P_{\max}$. Notice that

$$((a'_r h_{k,r}) b_{k,1} + (a_d h_{k,d}) b_{k,2})^2 \leq \tag{8}$$
$$((a'_r h_{k,r})^2 + (a_d h_{k,d})^2) \left(b_{k,1}^2 + b_{k,2}^2\right),$$

and the equality holds if and only if

$$\frac{b_{k,1}}{a'_r h_{k,r}} = \frac{b_{k,2}}{a_d h_{k,d}} = \rho_k. \tag{9}$$

With $|b_{k,1}|^2 + |b_{k,2}|^2 = P \leq P_{\max}$, ρ_k can be computed as $\rho_k(P) = \sqrt{P}/\sqrt{\left(a'_r h_{k,r}\right)^2 + \left(a_d h_{k,d}\right)^2}$, and $a'_r h_{k,r} b_{k,1} + a_d h_{k,d} b_{k,2}$ becomes [9]

$$\gamma_k(P) = \sqrt{P} \cdot \sqrt{(a'_r h_{k,r})^2 + (a_d h_{k,d})^2}. \tag{10}$$

In the basic method, with a_d fixed to a_0 and the initial $\mathbb{N}_r$, other parameters $a'_r, b_{k,1}, b_{k,2}$ will be computed by minimizing MSE_r in (4). This method only tries to fill the gap in the magnitude of misaligned signals, and is called CohR-ZF (zero-forcing). It is not necessarily optimal. In addition, $\gamma_k(P_{\max})$ may still be less than 1 because of the constraint of relay TP.

C. Node Scheduling

Now look back at the TP consumed at the relay for a node $k \in \mathbb{N}_r$ in (5). There are two cases to be considered.

1) Case (i): When relay r is close to node k while far from sink d, $h_{r,d}$ is small and $h_{k,r}$ is large, which leads to large TxR_k. The number of nodes that can be helped by the relay is small. So it is possible that not all nodes in $\mathbb{N}_r$ can be helped.

We compute the reduced MSE for x_k as the difference of MSE before and after using relay,

$$\triangle\mathrm{MSE}_k = |1 - a_0 h_{k,d} b_k|^2 - |1 - \min(\gamma_k(P_{\max}), 1)|^2. \tag{11}$$

The MSE for x_k after using relay is 0, if $\gamma_k(P_{\max})$ is no less than 1.0. The required relay TP TxR_k is computed by (5). Here both $\triangle\mathrm{MSE}_k$ and TxR_k are non-negative.

A flag I_k is used to denote the node scheduling. $I_k = 1$ indicates that node $k \in \mathbb{N}_r$ uses the relay, and 0 otherwise. It is expected that MSE is reduced to the largest extent with the limited relay TP. Then, the node scheduling problem in case (i) is defined as follows:

$$\operatorname*{argmax}_{I_k} \sum_{k \in \mathbb{N}_r} \triangle\mathrm{MSE}_k \cdot I_k, \tag{12}$$
$$\text{s.t.} \sum_{k \in \mathbb{N}_r} \mathrm{TxR}_k \cdot I_k \leq P_{\max}, I_k = 0, 1.$$

Then, all nodes with $I_k = 1$ in $\mathbb{N}_r$ forms a new $\mathbb{N}_r$.

This is an ILP (integer linear programming) problem [11], which is NP-hard. Here we consider a heuristic method.

Because relay TP is limited, it is expected that a large $\triangle\mathrm{MSE}_k$ is achieved at the cost of a small TxR_k. Hence, we compute

$$\xi_k = \frac{\triangle\mathrm{MSE}_k}{\mathrm{TxR}_k} \tag{13}$$

as the metric for evaluating the priority of a node using relay. Nodes with low priority will be removed from $\mathbb{N}_r$ to $\mathbb{N}_d$ if the overall relay TP, $\sum_{k \in \mathbb{N}_r} \mathrm{TxR}_k$, is greater than the constraint.

2) Case (ii): When relay r is close to sink d, but far from node k, $h_{r,d}$ is large and $h_{k,r}$ is small. So TxR_k tends to be small. In such cases, there is a surplus in relay TP after helping nodes whose signals are misaligned in the direct transmission. Nodes in $\mathbb{N}_d$ do not need help to reduce MSE, but the TP at nodes may be large. Then, the remaining relay TP can be used to reduce node TP. By using a small amount of relay TP, we wish to reduce more node TP. Because the relay TP is limited, it is also necessary to decide which nodes to move from $\mathbb{N}_d$ to $\mathbb{N}_r$ first.

By using relay, the TP saved at node k is the difference before and after using relay, $\triangle\mathrm{TxP}_k = b_k^2 - (b_{k,1}^2 + b_{k,2}^2) \geq 0$, while the TP consumed at relay for node k is TxR_k. Using I_k to indicate whether node $k \in \mathbb{N}_d$ uses the relay, node scheduling is defined as follows:

$$\operatorname*{argmax}_{I_k} \sum_{k \in \mathbb{N}_d} \triangle\mathrm{TxP}_k \cdot I_k, \tag{14}$$
$$\text{s.t.} \sum_{k \in \mathbb{N}_d} \mathrm{TxR}_k \cdot I_k \leq P_{\max} - \sum_{k \in \mathbb{N}_r} \mathrm{TxR}_k, \; I_k = 0, 1.$$

Then, all nodes with $I_k = 1$ in $\mathbb{N}_d$ are moved to $\mathbb{N}_r$.

This is also an ILP problem. Here, we consider a heuristic method, and compute

$$\eta_k = \frac{\triangle\mathrm{TxP}_k}{\mathrm{TxR}_k} \tag{15}$$

as the metric. Obviously, a node with larger η_k ($\eta_k > 1$) should be given higher priority to use the relay, unless the constraint of relay TP is reached.

The heuristic method to the ILP problem is not necessarily optimal. For the cases where $|\mathbb{N}_r|$ or $|\mathbb{N}_d|$ is no more than 16, we confirmed by brute force search that the heuristic metric in (13) is optimal with a probability 97.8% and the heuristic metric in (15) is optimal with a probability 74.6%.

D. Whole Algorithm

In order to use relay to reduce MSE more aggressively, it is necessary to adjust both a_d and a'_r. But separately adjusting a_d and a'_r may over reduce the noise and greatly increase node TP [9]. To focus on minimizing signal distortion, we choose to keep noise power fixed. In other words, $a_d^2 + (a'_r)^2 = a_0^2$. Then, when switching a node from direct transmission to using relay, its signal magnitude,

$$\gamma_k(P) = \sqrt{P} \cdot \sqrt{h_{k,r}^2 a_0^2 - (h_{k,r}^2 - h_{k,d}^2) a_d^2}, \tag{16}$$

is a decreasing function of a_d if $h_{k,r} > h_{k,d}$, which is the typical case. So signal distortion $|1 - \min(\gamma_k(P), 1)|$

This article has been accepted for publication in IEEE Wireless Communications Letters. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/LWC.2022.3189010

4

Algorithm 1 Find optimal parameters and compute MSE.

1: **procedure** FINDPARAMFORCOHR-NS($h_{k,r}$, $h_{k,d}$)
2:     Minimizing MSE in (7), get a_0
3:     Initializing, $a_d \leftarrow a_0$, $a'_r \leftarrow 0$, MSE $\leftarrow \infty$
4:     **while** $a_d > 0$ **do**   ▷ Iteration
5:         $\mathbb{N}_r \leftarrow \{k | a_d \cdot h_{k,d} \cdot \sqrt{P_{max}} < 1\}$
6:         $\mathbb{N}_d \leftarrow \{k | k \notin \mathbb{N}_r\}$
7:         Invoke ProcRelay($\mathbb{N}_r$, $h_{k,r}$, $h_{k,d}$, a'_r, a_d), get $b_{k,1}$
8:         Compute TxR$_k$ for $k \in \mathbb{N}_r$ by (5)
9:         **if** ($\sum_{k \in \mathbb{N}_r}TxR_k > P_{max}$) **then** ▷ Dec nodes in $\mathbb{N}_r$
10:             Compute ξ_k for $k \in \mathbb{N}_r$ by (13)
11:             Sort nodes in $\mathbb{N}_r$ in increasing order of ξ_k
12:             Move from $\mathbb{N}_r$ to $\mathbb{N}_d$ if $\sum_{k \in \mathbb{N}_r}TxR_k > P_{max}$
13:         **else** ▷ Inc nodes in $\mathbb{N}_r$ to reduce node TX power
14:             Compute η_k for $k \in \mathbb{N}_d$ by (15)
15:             Sort nodes in $\mathbb{N}_d$ in the decreasing order of η_k
16:             Move k from $\mathbb{N}_d$ to $\mathbb{N}_r$ if $\sum_{\{k\} \cup \mathbb{N}_r}TxR_k < P_{max}$
17:         **end if**
18:         MSE$_r$ = ProcRelay($\mathbb{N}_r$, $h_{k,r}$, $h_{k,d}$, a'_r, a_d)
19:         MSE$_d = \sum_{k \in \mathbb{N}_d} |1 - \min(a_d h_{k,d}\sqrt{P_{max}}, 1)|^2 + \sigma^2 |a_d|^2$
20:         MSE$'(a'_r, a_d)$ = MSE$_r$ + MSE$_d$
21:         **if** MSE$'(a'_r, a_d) <$ MSE **then**
22:             MSE$(a'_r, a_d) \leftarrow$ MSE$'(a'_r, a_d)$
23:             $a_d \leftarrow a_d - \delta$, $a'_r \leftarrow \sqrt{a_0^2 - a_d^2}$ ▷ Update a_d, a'_r
24:         **else**
25:             break
26:         **end if**
27:     **end while**
28:     **return** a'_r, a_d, MSE(a'_r, a_d)
29: **end procedure**
30: **procedure** PROCRELAY($\mathbb{N}_r$, $h_{k,r}$, $h_{k,d}$, a'_r, a_d)
31:     **for** $k \in \mathbb{N}_r$ **do**   ▷ Iteration on $\mathbb{N}_r$
32:         Compute $\gamma_k(P_{max})$ by (10)
33:         $\rho_k = \min(\gamma_k(P_{max}), 1) / \left((a'_r h_{k,r})^2 + (a_d h_{k,d})^2\right)$
34:         $b_{k,1} = \rho_k \cdot a'_r h_{k,r}$
35:         $b_{k,2} = \rho_k \cdot a_d h_{k,d}$
36:     **end for**
37:     MSE$_r = \sum_{k \in \mathbb{N}_r} |1 - \min(\gamma_k(P_{max}), 1)|^2 + \sigma^2 |a'_r|^2$
38:     **return** MSE$_r$
39: **end procedure**

---

decreases with a_d until it reaches 0. If node TP is large enough, magnitudes of signals in $\mathbb{N}_d$ remain 1.0, while the overall MSE decreases. But this requires larger node TP for the direct transmission, and more nodes will be removed from $\mathbb{N}_d$ to $\mathbb{N}_r$. Under the constraint of relay TP, some nodes cannot use the relay, and the overall MSE will increase again. Therefore, we can gradually decrease a_d from a_0 while increase a_r from 0, so that MSE may reach a minimum. Although this is not necessarily the global minimum, it is so under most cases, and in the case of a local minimum, its difference from the global minimum is small.

The whole process of finding optimal parameters and the corresponding computation MSE is described in Algorithm 1. With K nodes, the complexity is $O(K^2)$ in AirComp. In Algorithm 1, finding the initial value a_0 by AirComp (line 2)
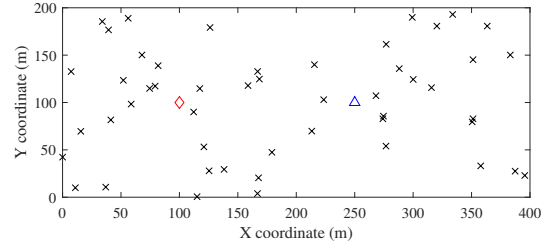

Fig. 2: Simulation scenario: 50 nodes ($\times$) randomly distributed in a $400m \times 200m$ area, 1 sink ($\diamondsuit$, (100, 100)) and 1 relay ($\triangle$, (250, 100)). Node deployment changes per evaluation.

takes the same computation. The computation costs of computing TxR (lines 7-8), decreasing nodes in $\mathbb{N}_r$ (lines 9-12) and moving nodes from $\mathbb{N}_d$ to $\mathbb{N}_r$ (lines 13-16) are $O(|\mathbb{N}_r|)$, $O(|\mathbb{N}_r|^2)$ and $O(|\mathbb{N}_d|^2)$, respectively. Here $|\cdot|$ represents the size of a set. The final computations of MSE$_r$ and MSE$_d$ (lines 18-19) are $O(|\mathbb{N}_r|)$ and $O(|\mathbb{N}_d|)$. Because $|\mathbb{N}_r| + |\mathbb{N}_d| = K$, the computation per iteration is approximately $O(K^2)$. Assume the number of iterations is N_{iter}, the overall computation cost is $N_{iter} \cdot O(K^2)$.

IV. SIMULATION EVALUATION

Here, we evaluate the proposed method (CohR-NS), comparing it with the AirComp method [6] that only exploits the direct link, the SimRelay (when a node uses a relay, its direct transmission is neglected), and CohRelay method in [9]. SimRelay and CohRelay are modified so that the overall relay TP is no more than the constraint. The basic method only involving Sec. III-A and III-B is named as CohR-ZF. The method that iterates over all possible a_d to find global minimal MSE is called CohR-Opt.

Fig. 2 shows the simulation scenario. 50 sensor nodes are randomly and uniformly distributed in a rectangle area ($400m \times 200m$). The frequency is set to 2.4GHz. A hybrid free-space/two-ray path loss model is used and path loss is 80dB at a distance of 90m. Each link experiences independent block Rayleigh fading (channel gains are the same in two slots). It is assumed that the relay-sink link does not experience fading. As for the power setting, $P_{max} = 15$dB, and $P_{max} \cdot \mathbb{E}\{|x_k|^2\}$ corresponds to 5dBm. When receiving signals, both sink d and relay r first amplify each signal to around the noise level (the strength of all signals is much greater than that of noise) for the A/D conversion and then amplify the signal in the digital domain to $\mathbb{E}\{|x_k|^2\} = \sigma^2 = 1$ without affecting signal to noise ratio. In the evaluation, we mainly use MSE and average power as the metrics. Average power is computed as the ratio of the overall power (TP of all nodes and the relay plus receive power of the relay, for which, it is assumed that the relay consumes the same power for receiving and transmission for simplicity) to the number of nodes. The simulation is run for 500 times in the Matlab environment and the average results are presented.

First we fix the sink position to (100, 100) and change the relay position along the line Y=100, from (125, 100) to (400, 100). The relay-sink distance changes from 25m to 300m accordingly. Fig. 3a shows the computation MSE in different
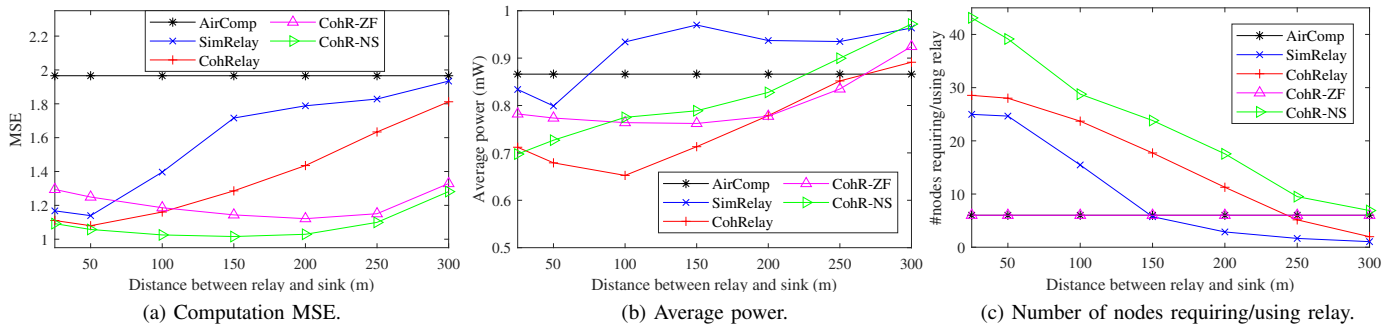
Fig. 3: Performance with respect to relay-sink distance. (c) is the number of nodes requiring relay in AirComp and using relay in other methods. In CohR-ZF the relay only helps nodes that require relay, so the curves of CohR-ZF and AirComp overlap.

methods. AirComp has the largest MSE, while SimRelay and CohRelay reduce MSE by using the relay. But it is obvious that their performance degrades when relay-sink distance increases. In comparison, CohR-ZF helps to reduce MSE in the long distance range, while CohR-NS further achieves the least MSE in the whole range. We also confirmed that the degradation of MSE in CohR-NS in comparison to CohR-Opt is only 0.05% and 0.28%, nearly optimal, when relay-sink distance is 150m and 200m, respectively.

There is a tradeoff between MSE and power consumption, and a low computation MSE is usually achieved at the cost of increased power. As shown in Fig. 3b, average power increases a little in CohR-NS, compared with that in CohRelay. But it is still less than that in AirComp in the typical distance ranges, because using coherent relay helps to reduce node TP.

Fig. 3c shows the number of nodes requiring or using relay in different methods. Unsurprisingly, when the relay-sink distance increases, more power is required for helping each node, and the number of nodes using relay decreases in all relay methods. But more nodes can use the relay in CohR-NS than in other methods. It is this aggressive relay policy that helps CohR-NS to reduce MSE.

Next we evaluate the impact of the number of nodes in the network. The relay-sink distance is fixed to 150m, but the number of nodes is changed from 20 to 500. Computation MSE and average power are summarized in Table I. It is clear that MSE increases with the number of nodes, because it becomes difficult to align signal magnitude when there are more nodes. As a result, signal magnitude actually decreases, so nodes with large channel gain can save power, and the average power decreases. Compared with CohRelay, MSE reduction in CohR-NS is more than twice the increase in average power, which confirms that the proposed method is effective in reducing MSE while suppressing the increase of power consumption.

V. Conclusion

AirComp as a promising data aggregation method for future sensor networks faces the reliability issue. To address this problem, this paper enhances the AF based relay method for AirComp and proposes a new method for node scheduling, considering the constraint of relay TP and node TP. Simulation evaluations confirm that the proposed method is more

TABLE I: Computation MSE and average power with respect to the number of nodes in the network.

#nodes	20	50	100	200	500
MSE (AirComp)	1.189	1.966	2.858	4.053	6.497
MSE (CohRelay)	0.816	1.286	1.896	2.809	4.670
MSE (CohR-NS)	0.631	1.016	1.447	2.071	3.298
Dec (vs CohRelay)	22.7%	21.0%	23.7%	26.3%	29.4%
Power (AirComp)	1.137	0.866	0.690	0.550	0.397
Power (CohRelay)	0.885	0.637	0.503	0.409	0.309
Power (CohR-NS)	0.927	0.691	0.564	0.463	0.354
Inc (vs CohRelay)	4.53%	8.31%	11.87%	12.87%	14.30%

effective than previous methods in reducing computation MSE meanwhile suppressing the increase of power consumption, and scales better with the number of nodes. In the future, we will further study the relay selection problem.

References

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

[2] M. Goldenbaum, H. Boche, and S. Staczak, "Nomographic functions: Efficient computation in clustered Gaussian sensor networks," *IEEE Transactions on Wireless Communications*, vol. 14, no. 4, pp. 2093–2105, 2015.

[3] O. Abari, H. Rahul, and D. Katabi, "Over-the-air function computation in sensor networks," *CoRR*, vol. abs/1612.02307, 2016. [Online]. Available: http://arxiv.org/abs/1612.02307

[4] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep over-the-air computation," in *IEEE Globecom Proceedings*, 2020.

[5] J. Xiao, S. Cui, Z. Luo, and A. J. Goldsmith, "Linear coherent decentralized estimation," *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 757–770, 2008.

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

[7] X. Cao, G. Zhu, J. Xu, and K. Huang, "Optimized power control for over-the-air computation in fading channels," *IEEE Transactions on Wireless Communications*, vol. 19, no. 11, pp. 7498–7513, 2020.

[8] S. Tang, P. Popovski, C. Zhang, and S. Obana, "Multi-slot over-the-air computation in fading channels," *CoRR*, vol. abs/2010.13559, 2021. [Online]. Available: https://arxiv.org/abs/2010.13559

[9] S. Tang, H. Yin, C. Zhang, and S. Obana, "Reliable over-the-air computation by amplify-and-forward based relay," *IEEE Access*, vol. 9, pp. 53 333–53 342, 2021.

[10] Y. Zhao, R. Adve, and T. J. Lim, "Improving amplify-and-forward relay networks: Optimal power allocation versus selection," in *2006 IEEE International Symposium on Information Theory*, 2006, pp. 1234–1238.

[11] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity*. New York: Dover Publications, 1998.