Multi-View Feature Transmission for Edge Inference

I. Abstract

II. Introduction

The breakthrough and advancement of recent artificial intelligence (AI) technology have enabled it to penetrate into a wide range of fields, ranging from computer vision [1], natural language processing [2] to autonomous driving [3]. AI is also envisioned to integrate with wireless network to shape the future of next-generation communication [4]–[6]. It is predicted that the future network will connect billions of edge devices (e.g. IoT device) [7] and generate more than 150 trillion gigabytes of data [**?**]. The massive data combined with continuously increasing computing power paves the way for deploying trained AI models and developing various intelligent applications at the network edge, which falls into a new research area named *edge inference* [8]–[10]. Specifically, edge inference pushes the inference processes of AI models to the network edge in close proximity to data sources and aims at finishing a specific inference task with accuracy and latency requirement. In contrast to the existing the communication system design that usually takes bit-error rate (BER) as the performance metrics, the edge inference more concerns how to achieve fast and accurate inference regardless of the transmission errors. This reveals a paradigm shift in current wireless system design strategy from the data-oriented communication to task-oriented communication.

There are currently considerable research efforts in the community dedicated to the efficient implementation of edge inference [11]–[15], where the split-inference is arguably regarded as the state-of-the-art inference architecture. Named after the task split, split inference splits computation-intensive part from edge devices to edge servers with powerful computing resources. The remaining part executed on edge devices is to extract and transmit only the task-relevant feature from raw data containing a large amount of redundant information to the edge server. Feature extraction methods for general purpose include principal component analysis (PCA) and multi-layer perceptrons (MLPs). If the edge devices are equipped with a custom neural

processing unit (NPU), such as the Apple's bionic chip A12 [16], CNN and RNN can also be considered for feature extraction of visual and time series data respectively.

III. System Model

We consider an edge inference system with K sensing devices (e.g., radar, camera) and a mobile device (e.g., autonomous vehicles). The sensing devices sense the circumjacent environment information as raw data, and the mobile devices are interested in leveraging the sensing data (features) generated by all sensing devices to perform a real-time inference task with as high inference performance as possible, e.g., classification. Specifically, in this paper, we consider a classification model as inference task. The related application scenarios for such an consideration include obstacles detection, mapping construction.

A. Feature Distribution Model

Let $\mathbf{x} \in \mathbb{R}^D$ denote ground-true sensory data of the source, where $\mathbf{x}$ is assumed to follows a mixture of Gaussian distributions with L classes. Therefore, its probability density function (PDF) should follow

$$f(\mathbf{x}) = \frac{1}{L}\sum_{\ell=1}^{L}\mathcal{N}(\boldsymbol{\mu}_\ell, \boldsymbol{\Sigma}), \tag{1}$$

where $\boldsymbol{\mu}_\ell \in \mathbb{R}^D$ is the centroid of the ℓ-th class, and $\boldsymbol{\Sigma} \in \mathbb{R}^{D\times D}$ is a covariance matrix and is the same for all classes. Due to different circumstances of the sensing devices (e.g., placement location, observing direction, etc), each of them can only sense a corrupted version of the ground-true data, which introduces different levels of distortion. As a result, the feature vector at the sensor k can be modeled as

$$\mathbf{x}_k = \mathbf{x} + \mathbf{e}_k, \tag{2}$$

where $\mathbf{e}_k$ is the sensing distortion. According to [17], the sensing distortion vector follows Gaussian distributions with mean zero and covariance $\mathbf{E}_k$, i.e.,

$$\mathbf{e}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{E}_k). \tag{3}$$

Further, we assume that the principal component analysis (PCA) method has been applied to each feature vector of the sensor before feature transmission, and thus the covariance matrix of the feature vector at sensor k can be seen as a diagonal matrix with diagonal element $\sigma_1^2 + e_1^2, \cdots, \sigma_D + e_d^2$. This not only reduces communication overhead, but also simplifies subsequent analysis by ensuring that different feature dimensions are independent of each other.

In fact, the mean $\boldsymbol{\mu}_\ell$ and the covariance $\boldsymbol{\Sigma}$ can be priorly estimated during the training phase, so they are known to the mobile device. However, in this paper, we only focus on the inference phase. At the begining of inference phase, the estimated principal component are broadcasted to all sensing devices. Each projects the sensed feature vector into the eigenspace and only the features processed by PCA will be transmitted.

B. Discriminant Gain

As mentioned before, mobile devices need to complete the classification tasks with as high inference performance as possible. However, due to the effects of quantization and channel noise, the features may suffer from varying degrees of distortion during transmission, which degrade the inference accuracy for classification tasks. To measure inference accuracy of model under different feature vectors, the discriminant gain [18] is used as objective function.

The discriminant gain is built on the well-known KL divergence. In particular, we first introduce the definition of discriminant gain in arbitrary class pair, which quantifies the ability of feature vector $\mathbf{x}$ to distinguish different classes. For the classes ℓ and ℓ', as the feature vector $\mathbf{x}$ follows the distribution in (1), its discriminant gain on $\mathbf{x}$ can be written as

$$\begin{aligned} G_{\ell,\ell'}(\mathbf{x}) &= \mathsf{D}_{KL}[\mathcal{N}(\boldsymbol{\mu}_\ell, \boldsymbol{\Sigma}) \parallel \mathcal{N}(\boldsymbol{\mu}_{\ell'}, \boldsymbol{\Sigma})] + \mathsf{D}_{KL}[\mathcal{N}(\boldsymbol{\mu}_{\ell'}, \boldsymbol{\Sigma}) \parallel \mathcal{N}(\boldsymbol{\mu}_\ell, \boldsymbol{\Sigma})] \\ &= (\boldsymbol{\mu}_\ell - \boldsymbol{\mu}_{\ell'})^{\mathsf{T}} \boldsymbol{\Sigma}^{-1} (\boldsymbol{\mu}_\ell - \boldsymbol{\mu}_{\ell'}) \\ &= \sum_{d=1}^{D} G_{\ell,\ell'}(\mathbf{x}(d)), \quad \forall(\ell, \ell'), \end{aligned} \tag{4}$$

where $x(d)$ is the d-th element of $\mathbf{x}$ and $G_{\ell,\ell'}(x(d))$ is given as

$$G_{\ell,\ell'}\left(\mathbf{x}(d)\right) = \frac{\left(\boldsymbol{\mu}_\ell(d) - \boldsymbol{\mu}_{\ell'}(d)\right)^2}{\sigma_d^2}, \quad 1 \le d \le D. \tag{5}$$

Then, the overall discriminant gain is defined as the average of all the pairwise discriminant gain, given as

$$\begin{aligned} G(\mathbf{x}) &= \frac{2}{L(L-1)} \sum_{\ell=1}^{L} \sum_{\ell<\ell'} G_{\ell,\ell'}\left(\mathbf{x}\right) \\ &= \frac{2}{L(L-1)} \sum_{\ell=1}^{L} \sum_{\ell<\ell'} \sum_{d=1}^{D} G_{\ell,\ell'}(\mathbf{x}(d)) \\ &= \sum_{d=1}^{D} G(\mathbf{x}(d)), \end{aligned} \tag{6}$$

where $G(\mathbf{x}(d))$ is the discriminant gain of the d-th feature elements, given as

$$G(\mathbf{x}(d)) = \frac{2}{L(L-1)} \sum_{\ell=1}^{L} \sum_{\ell<\ell'} \frac{\left(\boldsymbol{\mu}_\ell(d) - \boldsymbol{\mu}_{\ell'}(d)\right)^2}{\sigma_d^2}, \quad 1 \leq d \leq D. \tag{7}$$

C. Communication Model

To enable massive devices connection in edge inference over a large geographic area, we propose to leverage Cloud-RAN to support edge inference over ultra dense wireless networks. In Cloud-RAN, multiple RRHs are deployed close to the sensing devices and coordinated by a central processor (CP) to serve mobile devices cooperatively. In particular, we consider a Cloud-RAN consisting of one CP, M multi-antennas RRHs and K single-antenna sensing devices, where all RRHs are connected to the CP through noiseless finite-capacity fronthaul link. Further, let C_m denote the fronthaul capacity of the link between RRH m and the CP, which should satisfy an overall capacity constraint, i.e., $\sum_{m=1}^{M} C_m \leq C$. In the following, we specifically introduce featrue transmission process for edge inference.

In the feature transmission, we consider that all sensing devices communicate with the RRHs over a shared wireless multiple access channel via Aircomp. In this case, all sensing devices concurrently transmit their local features and enable the RRHs to directly receive an aggregated result of these local features. The RRHs quantize the received analog modulated signal and then forward the CP for decoding. By exploiting such the signal superposition property of a wireless multiple access channel, AirComp significantly improves the communication efficiency [19]–[21].

By leveraging Aircomp, the local feature encoded into the D-dimensional signal vector will be transmitted in D time solts and it is obvious to see that the design of Aircomp in all time slots are the same. Thus we only the feature transmission in d-th time slot in following sequel.

Let $s_k(d) = \mathbf{x}_k(d)$ denote the transmit signal in the d-th time slot, $b_k(d) \in \mathbb{C}$ denote the transmit pre-coded scalar of sensor k at the time slot d for the power control. Then, the signal received by RRH m can be given by

$$\mathbf{y}_m(d) = \sum_{k=1}^{K} \mathbf{h}_{k,m} b_k(d) s_k(d) + \mathbf{z}_m(d), \tag{8}$$

where $\mathbf{h}_{k,m} \in \mathbb{C}^N$ is the channel coefficient between device k and RRH m, N denotes the number of antennas on the RRH, and $\mathbf{z}_m(d) \sim \mathcal{CN}(0, \sigma_z^2 \mathbf{I})$ denotes the additive white Gaussian noise (AWGN) for RRH m.

To forward the received signals to the CP through the capacity-limited fronthaul links, RRH m need quantize the received signals then sends the quantized signals to the CP. In this paper, we use simple independent quantization scheme, where each RRH is assumed to perform signal quantization independently. The effect of quantization can be modeled as a test channel with the unquantized signals as the input and quantized signals as the output [22]. Further assuming that the test channel is Gaussian, the output signal is generated from the input signal corrupted by an additive Gaussian compression noise, i.e.,

$$\hat{\mathbf{y}}_m(d) = \mathbf{y}_m(d) + \mathbf{q}_m(d), \tag{9}$$

where $\mathbf{q}_m(d) \in \mathbb{C}^N \sim \mathcal{CN}(\mathbf{0}, \mathbf{Q}_m)$ denotes the quantization noise and $\mathbf{Q}_m$ is the diagonal covariance matrix of the quantization noise for RRH m due to independent quantization scheme.

The received signal at the CP from all RRHs is written as

$$\hat{\mathbf{y}}(d) = \left[\hat{\mathbf{y}}_1^{\mathsf{T}}(d), \cdots, \hat{\mathbf{y}}_M^{\mathsf{T}}(d)\right]^{\mathsf{T}} = \sum_{k=1}^{K} \mathbf{h}_k b_k(d) s_k(d) + \mathbf{z}(d) + \mathbf{q}(d), \tag{10}$$

where $\mathbf{h}_k = \left[\mathbf{h}_{k,1}^{\mathsf{T}}, \cdots, \mathbf{h}_{k,M}^{\mathsf{T}}\right]^{\mathsf{T}}$, $\mathbf{z}(d) = \left[\mathbf{z}_1^{\mathsf{T}}(d), \cdots, \mathbf{z}_M^{\mathsf{T}}(d)\right]^{\mathsf{T}}$, and $\mathbf{q}(d) = \left[\mathbf{q}_1^{\mathsf{T}}(d), \cdots, \mathbf{q}_M^{\mathsf{T}}(d)\right]^{\mathsf{T}}$. To decode the aggregation signal $\hat{s}(d)$, the CP applies receive beamforming vector to the received $\hat{\mathbf{y}}$'s and take the real part, i.e.,

$$\begin{aligned} \hat{s}(d) &= \Re\left(\mathbf{m}_d^{\mathsf{H}} \hat{\mathbf{y}}(d)\right) \\ &= \Re\left(\mathbf{m}_d^{\mathsf{H}} \sum_{k=1}^{K} \mathbf{h}_k b_k(d) s_k(d)\right) + \mathbf{n}(d), \end{aligned} \tag{11}$$

where $\mathbf{m}_d = \left[\mathbf{m}_{d,1}^{\mathsf{T}}, \cdots, \mathbf{m}_{d,M}^{\mathsf{T}}\right]^{\mathsf{T}} \in \mathbb{C}^{MN}$ is the receive beamforming vector at time slot d, $\mathbf{n}(d) = \Re\left(\mathbf{m}_d^{\mathsf{H}}\left(\mathbf{z}(d) + \mathbf{q}(d)\right)\right)$ is the equivalent uplink noise. Given $\mathbf{m}_d$, the equivalent uplink noise is distributed as $\mathbf{n}(d) \sim \mathcal{N}(0, \sigma^2)$ with the variance

$$\sigma^2 = \frac{1}{2}\mathbf{m}_d^{\mathsf{H}}\left(\sigma_z^2 \mathbf{I} + \mathbf{Q}\right)\mathbf{m}_d, \tag{12}$$

where $\mathbf{Q} = \mathrm{diag}\{\mathbf{Q}_1, \cdots, \mathbf{Q}_m\}$ is defined as the uplink covariance matix.

D. Zero-Forcing Procoding

In uplink transmission, we assume that each sensor is able to perfectly estimate the channels between itself and all RRHs through downlink pilot signaling, and the CP can obtain gobal channel state information (CSI) of all sensing devices through uplink feedback [23]. Besides, we also assume that the channels between the devices and the CP is blocking fading, where the

channel gains remain invariant within one coherence block. Based on the perfect CSI, we carry out well-known zero-forcing coding design for transmission scalar, which is given by

$$\mathbf{m}_d^{\mathsf{H}}\mathbf{h}_k b_k(d) = c_k(d), 1 \leq k \leq K. \tag{13}$$

where $c_k(d) \geq 0$ is a real number representing the receive signal strength from device k. Hence, the transmit scalar at sensor k can be derived as

$$b_k(d) = \frac{c_k(d)(\mathbf{m}_d^{\mathsf{H}}\mathbf{h}_k)^{\mathsf{H}}}{\left|\mathbf{m}_d^{\mathsf{H}}\mathbf{h}_k\right|^2}, \forall k \in [K]. \tag{14}$$

By substituing the feature vector in (2) and transmission scalar into $\hat{s}(d)$ in (11), the aggregation signal $\hat{s}(d)$ can be rewritten as

$$\begin{aligned} \hat{s}(d) &= \sum_{k=1}^{K} c_k(d)s_k(d) + n(d) \\ &= \sum_{k=1}^{K} c_k(d)\mathbf{x}_k(d) + \sum_{k=1}^{K} c_k(d)\mathbf{e}_k(d) + n(d). \end{aligned} \tag{15}$$

Remark 1. *According to* (15)*, the impact of uplink communication can be boiled down to introducing the equivalent communication noise* $\mathbf{n}(d)$*. Given zero-forcing precoding design, we can perfectly compensate the channel fading and thus minimize the impact caused by the channel distortion. As a result, the inference accuracy will also be significantly improved. The subsequent numerical experiments will confirm that the presence of noise will reduce the accuracy of model inference by affecting the distribution of aggregated signal, as presented in the following lemma.*

Lemma 1. *The distribution of the aggregation signal* $\hat{s}(d)$ *is given by*

$$\hat{s}(d) \sim \frac{1}{L}\sum_{\ell=1}^{L} \mathcal{N}\left(\hat{\boldsymbol{\mu}}_\ell(d), \hat{\sigma}_d^2\right), \quad 1 \leq d \leq D, \tag{16}$$

where the means $\{\hat{\boldsymbol{\mu}}_\ell(d)\}$ *and the variance* $\{\hat{\sigma}_d^2\}$ *are*

$$\begin{cases} \hat{\boldsymbol{\mu}}_\ell(d) = \sum_{k=1}^{K} c_k(d)\boldsymbol{\mu}_\ell(d), \\ \hat{\sigma}_d^2 = \left(\sum_{k=1}^{K} c_k(d)\right)^2 \sigma_d^2 + \sum_{k=1}^{K} c_k^2(d)e_d^2 + \sigma^2. \end{cases} \tag{17}$$

Proof. The proof is straightforward by substituting pdf of $\mathbf{x}_k(d)$ in (1), the sensing distortion distribution in (3) and uplink noise distribution. □

IV. Problem Formulation

In this section, we aim to maximize the total the discriminant gain of aggregation signal over all time slots, by jointly optimizing the user's transmit scalar at the devices given by the variables $\{c_k\}$, the quantization codebook $\mathbf{Q}$ at all RRHs, and the beamforming vector $\{\mathbf{m}_d\}$ at the CP, subject to the power constraints at the devices, the total fronthaul constraint from all RRHs, and the total energy constraint over all times slots.

A. Power Constraints

Limited by the transmitter, each device's transmit power should satisfy their own power constraint, i.e.,

$$\mathbb{E}\left[\left|b_k(d)s_k(d)^2\right|\right] = \left|b_k(d)\right|^2 \mathbb{E}\left[s_k(d)^2\right] \leq P_k, \forall k \in [K]. \tag{18}$$

Nonetheless, variance of the transmit signal $s_k(d)$, i.e., $\mathbb{E}\left[s_k(d)^2\right]$ is known by the CP as a prior information (e.g., estimated from the offline data samples). Therefore, the power constraint in (18) can be rewritten as

$$\left|b_k(d)\right|^2 \leq \hat{P}_k, \forall k \in [K], \tag{19}$$

where $\hat{P}_k = \frac{P_k}{\mathbb{E}[s_k(d)^2]}$ is the equivalent maximum transmit precoding power. Substituting the transmission scalar in (14) into the power constraint, we get

$$c_k(d)^2 \leq \hat{P}_k \left|\mathbf{m}_d^{\mathsf{H}} \mathbf{h}_k\right|^2, \forall k \in [K]. \tag{20}$$

B. Fronthaul Constraints

Recall that in uplink transmission, all RRHs independently quantize the received signal to satisfy the fronthaul link capacity constraints C. Based on the rate-distortion theory, the fronthaul

rates of M RRHs at the d-th time slot should satisfy

$$\begin{aligned}
&\sum_{m=1}^{M} C_m(d) = \sum_{m=1}^{M} I\left(\mathbf{y}_m(d); \hat{\mathbf{y}}_m(d)\right) \\
&= \sum_{m=1}^{M} \log \frac{\left|\sum_{k=1}^{K} \left|b_k(d)\right|^2 \mathbf{h}_{k,m}(\mathbf{h}_{k,m})^{\mathsf{H}} + \sigma_z^2 \mathbf{I} + \mathbf{Q}_m\right|}{|\mathbf{Q}_m|} \\
&\leq \sum_{m=1}^{M} \log \frac{\left|\hat{P}_k \sum_{k=1}^{K} \mathbf{h}_{k,m}(\mathbf{h}_{k,m})^{\mathsf{H}} + \sigma_z^2 \mathbf{I} + \mathbf{Q}_m\right|}{|\mathbf{Q}_m|} \\
&= \log \frac{\left|\hat{P}_k \sum_{k=1}^{K} \mathbf{h}_k(\mathbf{h}_k)^{\mathsf{H}} + \sigma_z^2 \mathbf{I} + \mathbf{Q}\right|}{|\mathbf{Q}|} \\
&\leq C
\end{aligned} \tag{21}$$

C. Energy Constraints

Herein we assume that the device have generated sensing data and some necessary processing steps have been performed. Hence, the energy consumption of each device mainly resulted from the data transmission, which is modeled as

$$\sum_{d=1}^{D} \sum_{k=1}^{K} \mathbb{E}\left[\left|b_k(d) s_k(d)^2\right|\right] = \sum_{d=1}^{D} \sum_{k=1}^{K} \left(\left|b_k(d)\right|^2 \mathbb{E}\left[s_k(d)^2\right] \cdot T\right) \leq E, \tag{22}$$

where E denote the total energy constraint, T is duration at each time slot and is the same for each slot. Likewise, substituting the transmission scalar in (14) into the power constraint again, the energy constraint can be rewritten as

$$\sum_{d=1}^{D} \sum_{k=1}^{K} \frac{\left|c_k(d)\right|^2}{\left|\mathbf{m}_d^{\mathsf{H}} \mathbf{h}_k\right|^2} \cdot \mathbb{E}\left[s_k(d)^2\right] \leq \frac{E}{T}. \tag{23}$$

D. Problem Formulation

Under the three kinds of constraints above, the inference-task-oriented problem can be formulated as

$$\mathscr{P}: \underset{\substack{\{c_k(d)\},\{\mathbf{m}_d\},\\ \mathbf{Q}}}{\text{maximize}} \quad G = \frac{2}{L(L-1)} \sum_{d=1}^{D} \sum_{\ell=1}^{L} \sum_{\ell<\ell'} \frac{\left(\hat{\boldsymbol{\mu}}_\ell(d) - \hat{\boldsymbol{\mu}}_{\ell'}(d)\right)^2}{\hat{\sigma}_d^2}$$

$$\text{subject to} \quad c_k^2(d) \le \hat{P}_k \left|\mathbf{m}_d^{\mathsf{H}} \mathbf{h}_k\right|^2, \forall k \in [K], \forall d \in [D], \tag{24a}$$

$$\sum_{d=1}^{D} \sum_{k=1}^{K} \frac{|c_k(d) s_k(d)|^2}{\left|\mathbf{m}_d^{\mathsf{H}} \mathbf{h}_k\right|^2} \le E, \tag{24b}$$

$$\log \frac{\left|\hat{P}_k \sum_{k=1}^{K} \mathbf{h}_k (\mathbf{h}_k)^{\mathsf{H}} + \sigma_z^2 \mathbf{I} + \mathbf{Q}\right|}{|\mathbf{Q}|} \le C. \tag{24c}$$

V. PROPOSED ALGORITHM

This section presents the details of the algorithm for solving this problem $\mathscr{P}$. We first transform the original problem $\mathscr{P}$ into an equivalent form through variables transformation. Then the equivalent problem can be decomposed into two sub-problems, which are solved by applying the successive convex approximation (SCA) and alternate convex search (ACS) techniques.

A. An Equivalent Problem

To simplify problem $\mathscr{P}$, we define

$$\boldsymbol{\alpha}(d) = \frac{2}{L(L-1)} \sum_{\ell=1}^{L} \sum_{\ell<\ell'} \frac{\left(\hat{\boldsymbol{\mu}}_\ell(d) - \hat{\boldsymbol{\mu}}_{\ell'}(d)\right)^2}{\hat{\sigma}_d^2}, \tag{25}$$

where $\boldsymbol{\alpha}(d)$ denotes the average discriminant gain on all class pairs of the d-th feature element. By substituting (17) into constraints (25), we have

$$\sum_{k=1}^{K} c_k^2(d) e_d^2 + \frac{1}{2} \mathbf{m}_d^{\mathsf{H}} \left(\sigma_z^2 \mathbf{I} + \mathbf{Q}\right) \mathbf{m}_d + \left(\sum_{k=1}^{K} c_k(d)\right)^2 \sigma_d^2 = \frac{\left(\sum_{k=1}^{K} c_k(d)\right)^2}{\boldsymbol{\alpha}(d)} \cdot \frac{2}{L(L-1)} \sum_{\ell=1}^{L} \sum_{\ell<\ell'} \left(\boldsymbol{\mu}_\ell(d) - \boldsymbol{\mu}_{\ell'}(d)\right)^2. \tag{26}$$

As a result, the original problem can be reformulated as

$$\begin{aligned} \underset{\substack{\{\boldsymbol{\alpha}(d)\},\{\beta_{k,d}\},\\ \{c_k(d)\},\mathbf{Q},\{\mathbf{m}_d\}}}{\text{maximize}} \quad & G=\sum_{d=1}^{D}\boldsymbol{\alpha}(d) \\ \text{subject to} \quad & \text{constraints (24a), (24b), (24c),} \end{aligned}$$

$$\sum_{k=1}^{K}c_k^2(d)e_d^2+\frac{1}{2}\mathbf{m}_d^{\mathsf{H}}\left(\sigma_z^2\mathbf{I}+\mathbf{Q}\right)\mathbf{m}_d+\left(\sum_{k=1}^{K}c_k(d)\right)^2\sigma_d^2\le\frac{\left(\sum\limits_{k=1}^{K}c_k(d)\right)^2}{\boldsymbol{\alpha}(d)}\cdot \tag{27a}$$

$$\frac{2}{L(L-1)}\sum_{\ell=1}^{L}\sum_{\ell<\ell'}\left(\boldsymbol{\mu}_\ell(d)-\boldsymbol{\mu}_{\ell'}(d)\right)^2.$$

Note that for problem (27), it can be easily showed that all constraints in (27a) should be satisfied with equality, since otherwise the objective value of problem (27) can be further increased by increasing (27a). Nevertheless, the problem is still very difficult to solve due to the high couple of the variables across multiple time slots in the constraints (24b). To make this problem feasible, let's introduce an auxiliary variable $\boldsymbol{\beta}=[\beta_{1,1},\beta_{1,2},\cdots,\beta_{k,d}]^{\mathsf{T}}$ as upper bound such that the following inequality holds

$$\frac{\left|c_k(d)s_k(d)\right|^2}{\left|\mathbf{m}_d^{\mathsf{H}}\mathbf{h}_k\right|^2}\le\beta_{k,d} \tag{28}$$

It can be showed that the second constraint can equivalently be written as

$$\frac{\left|c_k(d)s_k(d)\right|^2}{\beta_{k,d}}\le\left|\mathbf{m}_d^{\mathsf{H}}\mathbf{h}_k\right|^2,\forall k\in[K],\forall d\in[D], \tag{29a}$$

$$\sum_{d=1}^{D}\sum_{k=1}^{K}\beta_{k,d}\le E. \tag{29b}$$

Proof. Given a set of variables $\{c_k(d),\mathbf{m}_d^{\mathsf{H}}\}$ satisfying constraint (24b), let $\beta_{k,d}=\frac{\left|c_k(d)s_k(d)\right|^2}{\left|\mathbf{m}_d^{\mathsf{H}}\mathbf{h}_k\right|^2},\forall k\in[K],\forall d\in[D]$, then constraints (29a) (29b) holds. Given a set of variables $\{c_k(d),\mathbf{m}_d^{\mathsf{H}},\beta_{k,d}\}$ satisfying constraints (29a) (29b), then immediately constraint (28) holds by simple algebra operations. Simultaneously summing K and D on both sides of the inequality in constraint (28) and combining with (29b), the inequality (24b) is derived. □

Therefore, problem (27) is further reduced to

$$\begin{aligned} \underset{\substack{\{\boldsymbol{\alpha}(d)\},\{\beta_{k,d}\},\\ \{c_k(d)\},\mathbf{Q},\{\mathbf{m}_d\}}}{\text{maximize}} \quad & G=\sum_{d=1}^{D}\boldsymbol{\alpha}(d) \\ \text{subject to} \quad & \text{constraints (24a), (24c), (27a), (29a), (29b).} \end{aligned} \tag{30}$$

B. Alternating Optimization Approach

Algorithm 1 Proposed Algorithm for Solving Problem $\mathscr{P}$

Input: Initial points $\{\boldsymbol{\alpha}^{[0]}\}$, $\{c_k^{[0]}\}$, $\{\tilde{\mathbf{m}}_d^{[0]}\}$, $\mathbf{Q}^{[0]}$ and solution precision ϵ;

1: Set $t = 0$
2: **repeat**
3: Update $\{\boldsymbol{\alpha}^{[t]}(d)\}$, $\{c_k^{[t]}\}$, $\{\tilde{\mathbf{m}}_d^{[t]}\}$ by solving problem (38);
4: Update $\{\boldsymbol{\alpha}^{[t]}(d)\}$, $\{c_k^{[t]}\}$, $\mathbf{Q}^{[t]}$ by solving problem (39);
5: Calculate discirminant gain $t = t + 1$;
6: Set $t = t + 1$;
7: **until convergence**

Output: $\{\boldsymbol{\alpha}^{[t]}(d)\}$, $\{c_k^{[t]}\}$, $\{\tilde{\mathbf{m}}_d^{[t]}$, and $\mathbf{Q}^{[t]}$.

In this part, we propose an alternating optimization approach to solve problem (30) for obtaining a suboptimal solution. Specifically, we first fix the quantization matrix $\mathbf{Q}$ and optimize the other variables. With given $\mathbf{Q}$, problem (30) is reduced to the following problem:

$$\begin{aligned} \underset{\substack{\{\boldsymbol{\alpha}(d)\},\{\beta_{k,d}\},\\ \{c_k(d)\},\{\mathbf{m}_d\}}}{\text{maximize}} \quad & G = \sum_{d=1}^{D} \boldsymbol{\alpha}(d) \\ \text{subject to} \quad & \text{constraints (24a), (27a), (29a), (29b).} \end{aligned} \tag{31}$$

Although the objective function is convex, it is still challenging to solve problem (31) due to the non-convex constraints. In general , there is no standard method for solving such non-convex optimization problems optimally. Herein we adopt the successive convex approximation (SCA) technique to solve problem (31). To apply the SCA approach, we convert problem (31) from the complex domain to the real domain with the following variables:

$$\tilde{\mathbf{m}}_d = \left[\Re(\mathbf{m}_d)^{\mathsf{T}}, \Im(\mathbf{m}_d)^{\mathsf{T}}\right]^{\mathsf{T}}, \forall d \in [D], \tag{32a}$$

$$\tilde{\mathbf{H}}_k = \begin{bmatrix} \Re(\mathbf{h}_k\mathbf{h}_k^{\mathsf{H}}) & -\Im(\mathbf{h}_k\mathbf{h}_k^{\mathsf{H}}) \\ \Im(\mathbf{h}_k\mathbf{h}_k^{\mathsf{H}}) & \Re(\mathbf{h}_k\mathbf{h}_k^{\mathsf{H}}) \end{bmatrix}, \forall k \in [K], \tag{32b}$$

$$\tilde{\mathbf{Q}} = \begin{bmatrix} \Re(\tilde{\mathbf{Q}}) & -\Im(\tilde{\mathbf{Q}}) \\ \Im(\tilde{\mathbf{Q}}) & \Re(\tilde{\mathbf{Q}}) \end{bmatrix}. \tag{32c}$$

The problem (31) can be reformulated as follows:

$$\begin{aligned} \underset{\substack{\{\boldsymbol{\alpha}(d)\},\{\beta_{k,d}\},\\ \{c_k(d)\},\{\tilde{\mathbf{m}}_d\}}}{\text{maximize}} \quad & G=\sum_{d=1}^{D}\boldsymbol{\alpha}(d) \\ \text{subject to} \quad & \text{constraints (29b)}, \\ & c_k^2(d)\le \hat{P}_k\tilde{\mathbf{m}}_d^{\mathsf{T}}\tilde{\mathbf{H}}_k\tilde{\mathbf{m}}_d, \forall k\in[K],\forall d\in[D], \qquad (33a) \\ & \frac{|c_k(d)s_k(d)|^2}{\beta_{k,d}}\le \tilde{\mathbf{m}}_d^{\mathsf{T}}\tilde{\mathbf{H}}_k\tilde{\mathbf{m}}_d, \forall k\in[K],\forall d\in[D], \qquad (33b) \\ & \left(\sum_{k=1}^{K}c_k^2(d)e_d^2+\frac{1}{2}\tilde{\mathbf{m}}_d^{\mathsf{H}}\left(\sigma_z^2\mathbf{I}+\tilde{\mathbf{Q}}\right)\tilde{\mathbf{m}}_d+\sigma_d^2\right)\le \frac{\left(\sum_{k=1}^{K}c_k(d)\right)^2}{\boldsymbol{\alpha}(d)}\cdot \qquad (33c) \\ & \qquad\qquad \frac{2}{L(L-1)}\sum_{\ell=1}^{L}\sum_{\ell<\ell'}\left(\boldsymbol{\mu}_\ell(d)-\boldsymbol{\mu}_{\ell'}(d)\right)^2. \qquad (33d) \end{aligned}$$

We define

$$f_{k,d}(\tilde{\mathbf{m}}_d)=\tilde{\mathbf{m}}_d^{\mathsf{T}}\tilde{\mathbf{H}}_k\tilde{\mathbf{m}}_d, \forall k\in[K],\forall d\in[D], \qquad (34)$$

$$g_d\left(\{c_k\},\boldsymbol{\alpha}(d)\right)=\frac{\left(\sum_{k=1}^{K}c_k(d)\right)^2}{\boldsymbol{\alpha}(d)}, \forall d\in[D]. \qquad (35)$$

and then the following lemma is obtained.

Lemma 2. *Given the initial point* $\left\{\tilde{\mathbf{m}}_d^{[t]},c_k^{[t]},\boldsymbol{\alpha}^{[t]}\right\}$ *in the* t*-th iteration, the function* $\{f_{k,d}(\tilde{\mathbf{m}}_d)\}$, $\{g_d\left(\{c_k\},\boldsymbol{\alpha}(d)\right)\}$ *is lower bounded by their respective first-order Taylor expansion, i.e.,*

$$\begin{aligned} f_{k,d}(\tilde{\mathbf{m}}_d) &\ge \hat{f}_{k,d}(\tilde{\mathbf{m}}_d^{[t]}) \\ &= f_{k,d}(\tilde{\mathbf{m}}_d^{[t]})+\nabla_{\tilde{\mathbf{m}}_d}f_{k,d}(\tilde{\mathbf{m}}_d^{[t]})^{\mathsf{T}}(\tilde{\mathbf{m}}_d-\tilde{\mathbf{m}}_d^{[t]}) \qquad (36) \\ &= \left(2\tilde{\mathbf{H}}_k\tilde{\mathbf{m}}_d^{[t]}\right)^{\mathsf{T}}\tilde{\mathbf{m}}_d-(\tilde{\mathbf{m}}_d^{[t]})^{\mathsf{T}}\tilde{\mathbf{H}}_k\tilde{\mathbf{m}}_d^{[t]}, \forall k\in[K],\forall d\in[D], \end{aligned}$$

$$
\begin{aligned}
g_d\left(\{c_k\},\boldsymbol{\alpha}(d)\right) \geq\ & \hat{g}_d\left(\{c_k^{[t]}\},\boldsymbol{\alpha}^{[t]}(d)\right) \\
=\ & g_d\left(\{c_k^{[t]}\},\boldsymbol{\alpha}^{[t]}(d)\right)+\sum_{k=1}^{K}\nabla_{c_k(d)}g_d\left(\{c_k^{[t]}\},\boldsymbol{\alpha}^{[t]}(d)\right)\left(c_k(d)-c_k^{[t]}(d)\right) \\
& +\nabla_{\boldsymbol{\alpha}(d)}g_d\left(\{c_k^{[t]}\},\boldsymbol{\alpha}^{[t]}(d)\right)\left(\boldsymbol{\alpha}(d)-\boldsymbol{\alpha}^{[t]}(d)\right) \\
=\ & \frac{\left(\sum_{k=1}^{K}c_k^{[t]}(d)\right)^2}{\boldsymbol{\alpha}^{[t]}(d)}+\sum_{k=1}^{K}\frac{2\sum_{k=1}^{K}c_k^{[t]}(d)}{\boldsymbol{\alpha}^{[t]}(d)}\left(c_k(d)-c_k^{[t]}(d)\right) \\
& -\left(\frac{\sum_{k=1}^{K}c_k^{[t]}(d)}{\boldsymbol{\alpha}^{[t]}(d)}\right)^2\left(\boldsymbol{\alpha}(d)-\boldsymbol{\alpha}^{[t]}(d)\right),\forall d\in[D].
\end{aligned}
\tag{37}
$$

With any given local point $\left\{\tilde{\mathbf{m}}_d^{[t]},c_k^{[t]},\boldsymbol{\alpha}^{[t]}\right\}$ as well as the lower bounds, problem (33) is approximated as the following problem

$$
\begin{aligned}
\underset{\substack{\{\boldsymbol{\alpha}(d)\},\{\beta_{k,d}\},\\ \{c_k(d)\},\{\tilde{\mathbf{m}}_d\}}}{\text{maximize}}\quad & G=\sum_{d=1}^{D}\boldsymbol{\alpha}(d) \\
\text{subject to}\quad & \text{constraints (29b)}, \\
& c_k^2(d)\leq\hat{P}_k\hat{f}_{k,d}(\tilde{\mathbf{m}}_d^{[t]}),\forall k\in[K],\forall d\in[D], && \text{(38a)} \\
& \frac{|c_k(d)s_k(d)|^2}{\beta_{k,d}}\leq\hat{f}_{k,d}(\tilde{\mathbf{m}}_d^{[t]}),\forall d\in[D], && \text{(38b)} \\
& \left(\sum_{k=1}^{K}c_k^2(d)e_d^2+\frac{1}{2}\tilde{\mathbf{m}}_d^{\mathsf{H}}\left(\sigma_z^2\mathbf{I}+\tilde{\mathbf{Q}}\right)\tilde{\mathbf{m}}_d+\sigma_d^2\right)\leq\frac{2}{L(L-1)}\sum_{\ell=1}^{L}\sum_{\ell<\ell'} && \text{(38c)} \\
& \left(\boldsymbol{\mu}_\ell(d)-\boldsymbol{\mu}_{\ell'}(d)\right)^2\cdot\hat{g}_d\left(\{c_k^{[t]}\},\boldsymbol{\alpha}^{[t]}(d)\right),\forall d\in[D].
\end{aligned}
$$

As a result, this problem is convex, which can be efficiently solved by using convex optimization tools, e.g., CVX [24]. When fixing the receive beamforming vector $\{\mathbf{m}_d\}$, problem (30) is reduced to the following problem:

$$
\begin{aligned}
\underset{\substack{\{\boldsymbol{\alpha}(d)\},\{\beta_{k,d}\},\\ \{c_k(d)\},\{Q_m\}}}{\text{maximize}}\quad & G=\sum_{d=1}^{D}\boldsymbol{\alpha}(d) \\
\text{subject to}\quad & \text{constraints (24a), (24c), (27a), (29a), (29b)}.
\end{aligned}
\tag{39}
$$

It is not hard to verify that all constraints in (39) is convex respect to $\mathbf{Q}$. Again apply SCA to lower bound constraints (27a), then this problem also becomes convex. The proposed algorithm are summarized in Algorithm 1.

VI. NUMERICAL RESULTS

In this section, we evaluate the performance of the proposed AirComp-based edge inference system over Cloud-RAN.

A. Communication Setting

We consider a Cloud-RAN network setting with $K = 20$ devices and $M = 4$ RRHs, 4 antennas per RRH. The devices and RRHs are both randomly and independently located in a circular area with inner radius of $100m$ and outer radius of $500m$. The channel is modeled as the small-scale fading coefficients multiplied by the square root of the path loss, i.e., $\mathbf{h}_{k,m} = 10^{-pl(d)/20}\mathbf{s}_{k,m}$, where $pl(d)$ is the path loss in dB given as $30.6 + 36.7\log_{10}(d)$ and d (in meter) is the distance between the device k and RRH m. The small-scale fading coefficients $\mathbf{s}_{k,m}$ is assumed to follow the standard complex Gaussian distribution, i.e., $\mathbf{s}_{k,m} \sim \mathcal{CN}(0, \mathbf{I})$. The power spectral density of the background noise at each RRH is set as -169 dBm/Hz and the noise figure is 7 dB. All numerical results are averaged over 50 trails.

B. Inference Setting

1) *Inference Dataset*: We perform the inference task on the human motion datasets proposed in [25] where sensing simulator is adopted to produce various high-fidelity human motions. In our simulation, we only consider to identify four distinct human motion, i.e., *child walking, child pacing, adult pacing* and *adult walking*. The entire dataset contains 8000 samples, 2000 for each class. Following the setting in [26], the heights of children and adults are assumed to be uniformly distributed in interval [0.9m, 1.2m] and [1.6m, 1.9m], respectively. The speed of standing, walking, and pacing are 0 m/s, $0.5H$ m/s, and $0.25H$ m/s, respectively, where H is the height value. In addition, the heading of the moving human is set to be uniformly distributed in $[-180°, 180°]$.
2) *Inference Model*: SVM and MLP neural networks are two commonly used classification models in machine learning and they are considered for our inference task. Specifically, the neural network model consists of two hidden layer, each with 80 and 40 neurons. The human motion datasets is divided into non-overlapping training and test sets with 6400 and 1600 samples, respectively. Both models are trained using the same training data regardless of channel and data distortion.

C. Convergence of the Proposed Algorithm

In the subsection, we show the convergence behavior of the proposed algorithm and outline the relationships between the discriminant gain and inference accuracy. In Fig. 1, we plot the discriminant gain achieved by the proposed algorithm with power constraint $P = 23$ dBm. It is observed that the discriminant gain is able to increases quickly and converges in a few iterations, which demonstrates the effectiveness of the proposed algorithm for joint optimization. We also illustrate the effect of different discriminant gain values on the inference accuracy in Fig. 2. It can be found from the figure that no matter which inference model is used, the inference accuracy will increase as the discriminant gain grows. However, we also noticed that when the discriminant gain increased to a certain value, the inference accuracy remained almost unchanged even though the discriminant gain value continued to improve. The main reason for this result may be limited wireless resource constraint. When the allocation of limited wireless resources is close to the optimum, the optimization for objective function will not bring about a significant increase in accuracy. In addition, in terms of inference accuracy, the SVM is better than neural network models. This is because the training of the latter is overfitting, resulting in a complex model on a simple dataset.

D. Impact of Key System Parameters

In the subsection, we show the performance gain of joint optimization over other baseline methods under the wireless and fronthaul resource constraints, and investigate the impact of various key system parameters. For ease of presentation, we refer to our proposed algorithm for jointly optimizing transmit precoding, quantization noise matrix and receive beamforming as **Joint Optimization** and set following schemes as baselines for comparison:

- **Baseline 1: Uniform quantization with joint optimization of transmit precoding and receive beamforming.** In Baseline 1, the optimized portion of transmit precoding and receive beamforming follows Algorithm 1. The CP perform uniform quantization at all antennas across all RRHs, i.e., setting $\mathbf{Q} = \lambda \mathbf{I}$, where scalar λ can be easily be selected by binary search to exactly satisfies the capacity constraint (24a). The transmit precoding and receive beamforming are jointly optimized
- **Baseline 2: Uniform receive beamforming with joint optimization of transmit precoding and quantization matrix.** In Baseline 2, the optimized portion of transmit precoding and

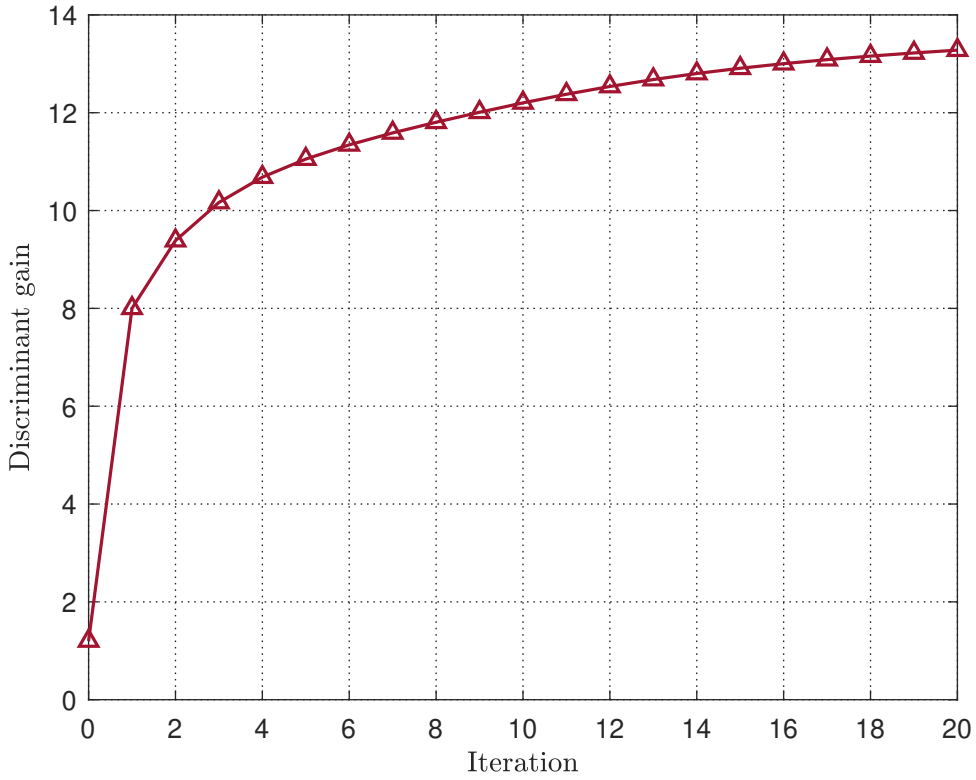


Figure 1: Convergence behavior of Proposed Algorithm.

quantization matrix follows Algotithm 1. The received beamforming are uniformly designed, i.e., setting $\mathbf{m}_d = \mathbf{1}$.

1) *Inference accuracy v.s. fronthaul capcity:* The inference accuracy of both models achieved by different schemes under various fronthaul capacity C is shown in Fig. 4. It can be observed that when fronthaul capacity increases in both cases, inference accuracy in all schemes is improved. And our proposed joint optimization achieves best performance over Baselines 1-2. Furthermore, it is also noticed that Baseline 1 with uniform quantization is consistently outperform Baseline 2. This indicates that the performance gain of joint optimization is mainly obtained by beamforming optimization on the CP.
2) *Inference accuracy v.s. energy:* Fig. 4 shows the inference accuracy of the both models achieved by different schemes under different energy constraints. From the figure, the inference accuracy increases as the energy requirement is gradually relaxed. This is due to the fact that more energy suppresses the channel noise and thus the discriminant gain is

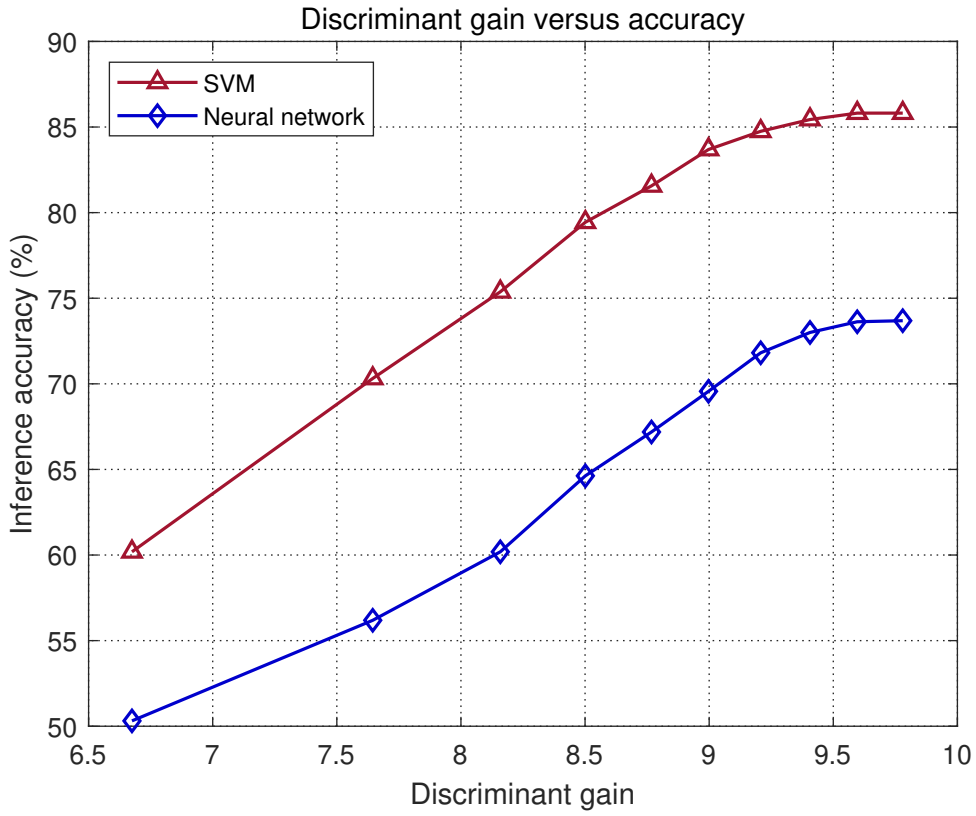
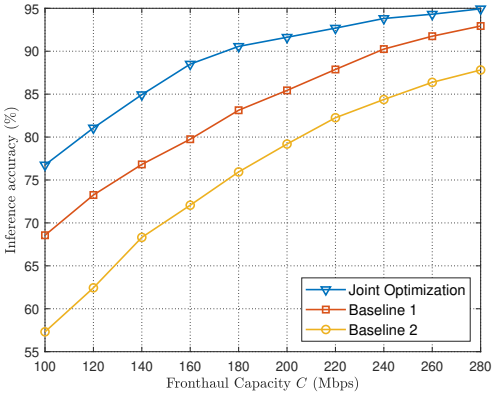
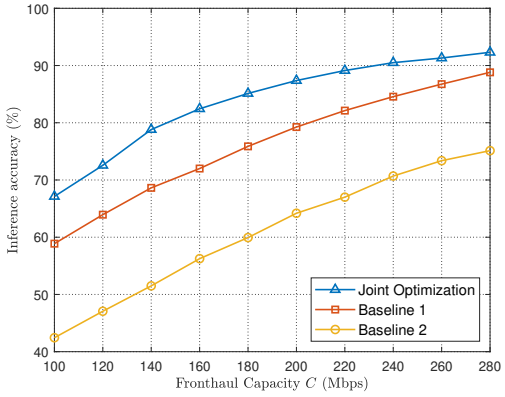


Figure 2: Inference accuracy versus discriminant gain

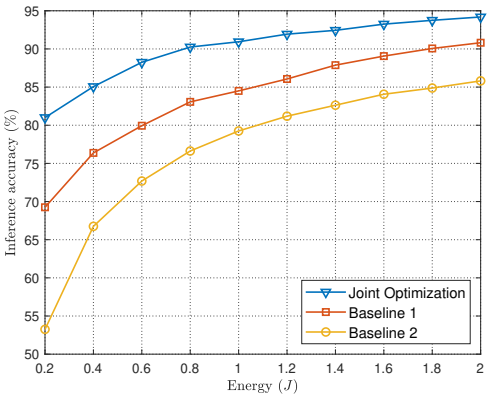


(a) Inference accuracy with SVM versus fronthaul capacity.

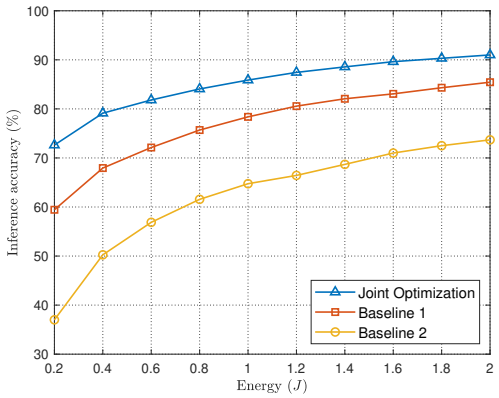


(b) Inference accuracy with MLP versus fronthaul capacity.

Figure 3: Inference accuracy comparison among different models under different fronthaul capacity.



(a) Inference accuracy with SVM versus energy.



(b) Inference accuracy with MLP versus energy.

Figure 4: Inference accuracy comparison among different models under different energy constraint.

enhanced. In addition, similar to the case of the fronthaul capacity, we can also conclude that Baseline 1 outperforms Baseline 2.

VII. CONCLUSION

REFERENCES

[1] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

[2] M. M. Lopez and J. Kalita, "Deep learning applied to nlp," *arXiv preprint arXiv:1703.03091*, 2017.

[3] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A survey of autonomous driving: Common practices and emerging technologies," *IEEE access*, vol. 8, pp. 58443–58469, 2020.

[4] K. B. Letaief, Y. Shi, J. Lu, and J. Lu, "Edge artificial intelligence for 6g: Vision, enabling technologies, and applications," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 5–36, 2021.

[5] K. B. Letaief, W. Chen, Y. Shi, J. Zhang, and Y.-J. A. Zhang, "The roadmap to 6g: Ai empowered wireless networks," *IEEE communications magazine*, vol. 57, no. 8, pp. 84–90, 2019.

[6] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Toward an intelligent edge: Wireless communication meets machine learning," *IEEE communications magazine*, vol. 58, no. 1, pp. 19–25, 2020.

[7] M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis, and E. K. Markakis, "A survey on the internet of things (iot) forensics: Challenges, approaches, and open issues," *IEEE Communications Surveys Tutorials*, vol. 22, no. 2, pp. 1191–1221, 2020.

[8] Y. Shi, K. Yang, T. Jiang, J. Zhang, and K. B. Letaief, "Communication-efficient edge ai: Algorithms and systems," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 4, pp. 2167–2191, 2020.

[9] J. Shao and J. Zhang, "Communication-computation trade-off in resource-constrained edge inference," *IEEE Communications Magazine*, vol. 58, no. 12, pp. 20–26, 2020.

[10] J. Shao, Y. Mao, and J. Zhang, "Task-oriented communication for multi-device cooperative edge inference," *IEEE Transactions on Wireless Communications*, 2022.

[11] D. Wen, X. Jiao, P. Liu, G. Zhu, Y. Shi, and K. Huang, "Task-oriented over-the-air computation for multi-device edge ai," *arXiv preprint arXiv:2211.01255*, 2022.

[12] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[13] K. Yang, Y. Shi, W. Yu, and Z. Ding, "Energy-efficient processing and robust wireless cooperative transmission for edge inference," *IEEE internet of things journal*, vol. 7, no. 10, pp. 9456–9470, 2020.

[14] X. Huang and S. Zhou, "Dynamic compression ratio selection for edge inference systems with hard deadlines," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8800–8810, 2020.

[15] E. Li, L. Zeng, Z. Zhou, and X. Chen, "Edge ai: On-demand accelerating deep neural network inference via edge computing," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 447–457, 2019.

[16] C.-J. Wu, D. Brooks, K. Chen, D. Chen, S. Choudhury, M. Dukhan, K. Hazelwood, E. Isaac, Y. Jia, B. Jia, *et al.*, "Machine learning at facebook: Understanding inference at the edge," in *2019 IEEE international symposium on high performance computer architecture (HPCA)*, pp. 331–344, IEEE, 2019.

[17] J. J. Xiao, S. Cui, Z. Q. Luo, and A. J. Goldsmith, "Power scheduling of universal decentralized estimation in sensor networks," *IEEE Transactions on Signal Processing*, vol. 54, no. 2, pp. 413–422, 2006.

[18] Q. Lan, Q. Zeng, P. Popovski, D. Gündüz, and K. Huang, "Progressive feature transmission for split inference at the wireless edge," *arXiv e-prints*, 2021.

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

[20] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 491–506, 2019.

[21] X. Li, G. Zhu, Y. Gong, and K. Huang, "Wirelessly powered data aggregation for iot via over-the-air function computation: Beamforming and power control," *IEEE Transactions on Wireless Communications*, vol. 18, no. 7, pp. 3437–3452, 2019.

[22] M. Peng, C. Wang, V. Lau, and H. V. Poor, "Fronthaul-constrained cloud radio access networks: Insights and challenges," *IEEE Wireless Communications*, vol. 22, no. 2, pp. 152–160, 2015.

[23] X. Cao, G. Zhu, J. Xu, and K. Huang, "Cooperative interference management for over-the-air computation networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 4, pp. 2634–2651, 2021.

[24] I. CVX Research, "CVX: Matlab software for disciplined convex programming, version 2.0." http://cvxr.com/cvx, Aug. 2012.

[25] G. Li, S. Wang, J. Li, R. Wang, X. Peng, and T. X. Han, "Wireless sensing with deep spectrogram network and primitive based autoregressive hybrid channel model," in *2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 481–485, IEEE, 2021.

[26] P. Liu, G. Zhu, S. Wang, W. Jiang, W. Luo, H. V. Poor, and S. Cui, "Toward ambient intelligence: Federated edge learning with task-oriented sensing, computation, and communication integration," 2022.