Minimizing End-to-End Latency for Joint Source-Channel Coding Systems

Kaiyi Chi*, Qianqian Yang†, Yuanchao Shu*, Zhaohui Yang†, Zhiguo Shi†
*College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China
†College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310027, China
E-mail: {kaiyichi, qianqianyang20, ycshu, yang_zhaohui, shizg}@zju.edu.cn

***Abstract*—While existing studies have highlighted the advantages of deep learning (DL)-based joint source-channel coding (JSCC) schemes in enhancing transmission efficiency, they often overlook the crucial aspect of resource management during the deployment phase. In this paper, we propose a resource management approach to minimize the transmission latency in an uplink JSCC-based system. We first analyze the correlation between end-to-end latency and task performance, based on which the end-to-end delay model for each device is established. Then, we formulate a non-convex optimization problem aiming at minimizing the maximum end-to-end latency across all devices, which is proved to be NP-hard. We then transform the original problem into a more tractable one, from which we derive the closed form solution on the optimal compression ratio, truncation threshold selection policy, and resource allocation strategy. We further introduce a heuristic algorithm with low complexity, leveraging insights from the structure of the optimal solution. Simulation results demonstrate that both the proposed optimal algorithm and the heuristic algorithm significantly reduce end-to-end latency. Notably, the proposed heuristic algorithm achieves nearly the same performance to the optimal solution but with considerably lower computational complexity.**

***Index Terms*—Semantic communication, joint source-channel coding, resource allocation, latency optimization.**

I. Introduction

Existing communication systems are developed based on Shannon's separation theorem, in which source coding and channel coding are separate steps. Source coding focuses on eliminating source redundancy, while channel coding introduces redundant information to enhance resilience against channel noises. While this separation is theoretically optimal with an infinitely large block length in memory-less channels [1], practical implementations often involve finite block lengths. Recently, the emergence of deep learning (DL) in other fields [2] has prompted researchers to explore joint source-channel coding (JSCC) schemes using DL techniques. DL-based JSCC models designed for text [3], image [4], speech [5] transmission, etc., have demonstrated significant advantages over their separated counterparts.

Traditional communication systems primarily focus on maximizing the transmission quality like bit error rate (BER) in resource allocation. However, this approach is not suitable for JSCC systems, where the emphasis is on the application performance rather than the quality of bit transmission. Despite this, the majority of research efforts focus on developing Deep Learning (DL)-based JSCC models, often overlooking the resource allocation challenge during system deployment. Some strides have been taken to address resource allocation strategies in DL-based JSCC systems in existing literature [6]–[8]. In [6], the authors tackled the resource allocation problem in downlink text transmission. They maximized the defined metric of semantic similarity (MSS) by jointly optimizing the transmission of semantic information and selecting resource blocks. In [7], the authors introduced the semantic transmission rate (S-R) and semantic spectral efficiency (S-SE), proposing to maximize the overall S-SE in an uplink scenario. Lastly, in [8], the authors delved into the resource allocation challenges in the downlink non-orthogonal multiple access (NOMA) system.

The aforementioned works focus on optimizing the weighted task performance of all devices. However, in some latency sensitive scenarios, we prefer the end-to-end delay of the device to be as small as possible. In addition, with the increasing size of the neural networks and the limited computation resources, the computational latency should also be taken into consideration. Motivated by these considerations, this paper aims to minimize the maximum end-to-end latency of the uplink transmission from all devices in the system while ensuring task performance.

We begin by theoretically analyzing the delay model to understand the relationship between task performance and end-to-end latency, which yields an end-to-end delay model for each device. We then formulate an optimization problem aimed at minimizing the maximum latency across all devices while ensuring task performance. This involves jointly considering the selection of compression ratio, channel truncation threshold, and the allocation of communication and computation resources. Recognizing the NP-hard nature of the problem, we employ problem transformation to make it more tractable. We obtain a closed-form solution for the optimal compression ratio, channel truncation threshold selection strategy, and resource allocation policy. Furthermore, we propose a heuristic algorithm with low complexity to tackle the problem in practical considerations. Simulation results validate the effectiveness of the proposed methods in terms of minimizing the end-to-end uplink latency.

The rest of the paper is organized as follows. Section II

This work is partly supported by NSFC under grant No. 62293481, No. 62201505, partly by the SUTD-ZJU IDEA Grant (SUTD-ZJU (VP) 202102).

introduces the system model and formulates the problem. Section III presents the proposed solution, followed by simulation results in Section IV. Section V concludes the paper.

II. System Model and Problem Formulation

In this section, we first introduce the DL-based JSCC system and establish the end-to-end latency model which takes both the communication and computation latency into account. Based on this, we then formulate an optimization problem to minimize the maximum latency among all devices while guaranteeing the task performance of each device.

A. System Model

Consider an uplink cellular network with a base station (BS) and a set $\mathcal{K} = \{1, 2, ..., K\}$ of K devices, as shown in Fig. 1. Each device aims to accomplish a specific task. We assume image transmission in this paper for the simplicity of analysis. However, we note that our proposed method can be extended into other transmission tasks, such as text transmission or speech transmission. We adopt the convolutional neural networks (CNN) based joint source-channel coding (DeepJSCC) network proposed in [4], where the encoder is deployed at the local device and the decoder is deployed at the edge server connected with the BS. We note that it can be extended to other types of networks such as DNN with similar analysis. The considered system operates as follows. Each device sends its information about channel state information (CSI), performance constraint, as well as the available local computation resource to the BS. Then, the BS will determine the communication and computation resource allocation policy of each device. After that, each device will compress the image with a specific JSCC network and transmit the extracted symbols to the BS via physical channel. Then, the BS decodes the content of received symbols using corresponding decoder network in parallel.

B. Encoding at Local Device

Each device uses an encoder locally to compress its source image of size $D_0 = 3 \times H \times W$, where H and W are the height and width, respectively. We define the compression ratio o_k as the proportion of the number of the transmitted symbols versus to the total number of symbols in the input image of device k. We have N predefined compression ratios (CRs), i.e., $o_k \in \{c_1, c_2, ..., c_N\}, \forall k \in \mathcal{K}$, and each CR corresponds to a specific encoder and decoder. We denote C_k^l as the computational cost per image of device k during local processing. We assume the images of all devices have the same resolution, i.e., H and W. According to [9], the computational cost of a CNN is proportional to the size of input resolution, i.e., HW. Thus, the computational complexity of the encoder at local device k is $L_k C_k^l = L_k C^s HW$, where L_k is the number of images to be processed at device k. C^s is the required number of CPU cycles per pixel using the encoder, which is determined by the architecture of the encoder network, i.e., CR value. We note that the C^s is almost the same for different encoders with different CRs in our test since they only differ in the number of feature maps at the output layer. Therefore, C^s is obtained through average under different CRs. Given that the local CPU-cycle frequency at the local device k is f_k^l, the computational latency of the encoding process at device k is

$$t_k^l = \frac{L_k C_k^l}{f_k^l}, \forall k \in \mathcal{K}. \tag{1}$$

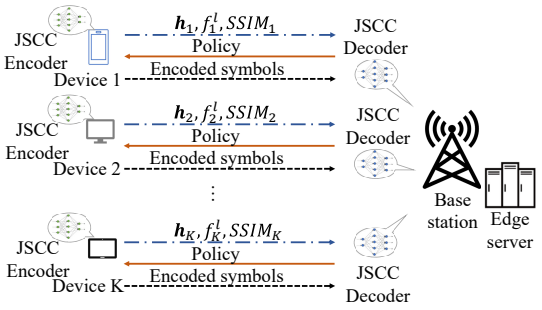

Fig. 1. The considered JSCC system model.

C. Transmission Model

We assume that time-division multiple access (TDMA) is applied for the channel access. Hence, each time frame is slotted and we denote the time slot allocated to device k for transmission per unit time as τ_k. We assume orthogonal frequency division multiplexing (OFDM) modulation where the whole bandwidth B is divided into M orthogonal sub-channels. At t-th time-slot, the received symbol from device k at the m-th sub-carrier is given by [10]

$$y_k^m(t) = r_k^{-\frac{\alpha}{2}} h_k^m(t) p_k^m(t) x_k^m(t) + z(t), \tag{2}$$

where $r_k^{-\frac{\alpha}{2}}$ denotes the path-loss of the link between the BS and device k, r_k denotes the distance between them and α is the path-loss exponent. $h_k^m(t)$ denotes the small scale fading of the channel following Rayleigh fading $\mathcal{CN}(0, 1)$, which is identically and independently distributed (i.i.d.) over k, m, t. $p_k^m(t)$ is the power allocated to the device k on the m-th sub-carrier over t-th time slot. $z(t)$ is the Gaussian channel noise with power of σ^2. For ease of notation, we will omit the index t in the following.

Following [10], we let the power allocated to each sub-carrier p_k^m adapt to the channel coefficient h_k^m to achieve the signal-to-noise ratio (SNR) alignment at the BS. Due to the limited power at the devices [11], we assume each device is subject to a long-term transmission power constraint $\mathbb{E}\left[\sum_{m=1}^{M} |p_k^m|^2\right] \leq P_k, \forall k \in \mathcal{K}$, where P_k is the maximum transmission power of the device k. Since channel coefficients are identically and independently distributed over different sub-channels, the above power constraint can be reformulated as $\mathbb{E}\left[|p_k^m|^2\right] \leq \frac{P_k}{M}, \forall k \in \mathcal{K}$.

We assume the knowledge of perfect CSI at each device. The device can thus perform power control on each sub-carrier to allow the received signals at the BS have the same amplitude across different subcarriers. Besides, to cope with deep fades, we adopt a more practical truncated channel inversion. To be more specific, a sub-channel will be cutoff for a device at this

time slot if its channel coefficient is less than a threshold g_k, i.e.,

$$p_k^m = \begin{cases} \frac{\sqrt{\rho_k}}{r_k^{-\frac{\alpha}{2}} h_k^m}, & |h_k^m|^2 \geq g_k, \\ 0, & |h_k^m|^2 < g_k, \end{cases} \tag{3}$$

where ρ_k is a scaling factor in order to meet the power constraint, and its also the power of the received symbols transmitted by the device k.

Since the channel coefficients follow Rayleigh distribution $\mathcal{CN}(0,1)$, the channel gain of the k-th link on the m-th sub-carrier $|h_k^m|^2$ follows the exponential distribution with unit mean. With truncated power allocation, we have $\mathbb{E}\left[|p_k^m|^2\right] = \frac{\rho_k}{r_k^{-\alpha}} \int_{g_k}^{\infty} \frac{1}{g} \exp(-g) dg \leq \frac{P_k}{M}$. Thus, the maximum received power of the symbols transmitted by the k-th device is bounded by

$$\rho_k \leq \frac{P_k}{M r_k^{\alpha} \operatorname{Ei}(g_k)}, \forall k \in \mathcal{K}, \tag{4}$$

where $\operatorname{Ei}(g_k) = \int_{g_k}^{\infty} \frac{1}{g} \exp(-g) dg$.

Besides the receive SNR, channel truncation ratio is also affected by the power-cutoff threshold g_k. We denote the percentage of the channels that are not truncated as the activate ratio ζ_k. When the number of transmitted symbols of device k is large enough, the activate ratio is equal to the probability that the channel gain is above the power-cutoff threshold, i.e., $\zeta_k = \Pr\left(|h_k|^2 > g_k\right) = e^{-g_k}, \forall k \in \mathcal{K}$. Hence, the number of expected activate channels per time slot is Me^{-g_k}. Denote the symbol duration of an OFDM symbol by T_s. Thus, the transmission delay of device k is given by

$$t_k^t = \frac{L_k D_0 o_k}{M e^{-g_k}} \frac{T_s}{\tau_k}, \forall k \in \mathcal{K}. \tag{5}$$

D. Decoding at the BS

The BS decodes the messages transmitted from the devices in parallel. We denote that the total computation resource of the edge server by F^c, and f_k^c as the computation resource allocated to decode the message from device k, which satisfies $\sum_k f_k^c \leq F^c$. Similarly, we denote C_k^d as the computational cost to decode an image at the decoder of device k, and $C^{s'}$ is the computational cost per pixel. Thus, the computational complexity for decoding message from device k at the decoder is $L_k C_k^d = L_k C^{s'} HW$. Thus, the computational latency of decoding message from device k at the edge is given as

$$t_k^c = \frac{L_k C_k^d}{f_k^c}, \forall k \in \mathcal{K}. \tag{6}$$

Hence, the end-to-end latency of encoding, transmitting, and then decoding message from device k can be given as $t_k = t_k^l + t_k^t + t_k^c, \forall k \in \mathcal{K}$.

E. Performance Metrics

In the considered system, the BS needs to take the performance on the uplink transmission of each device into consideration. We adopt the structure-similarity-index-measure (SSIM) as the performance metrics to evaluate the considered image transmission task as it can capture the perceived visual quality of the images well. According to the [12], the SSIM of the reconstructed image is determined by both the SNR of the received signal and the compression ratio. The SSIM by the adopted JSCC scheme increases as the compression ratio and SNR as shown in Fig. 2, where the simulation settings are shown in Section IV. We then use a generalized logistic function to fit the curve, which is given by

$$SSIM(o,\gamma) = A_{o,1} + \frac{A_{o,2} - A_{o,1}}{1 + e^{-(C_{o,1}\gamma + C_{o,2})}}, \tag{7}$$

where γ is the SNR of the device, and $A_{o,1}$, $A_{o,2}$, $C_{o,1}$, $C_{o,2}$ are constant values when the compression ratio o is given.

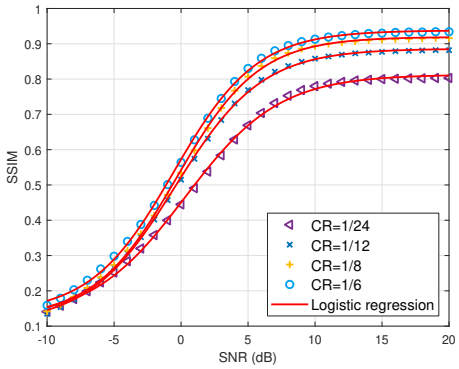

Fig. 2. Average SSIM of the reconstructed images vs. SNR under different compression ratios on ImageNet dataset.

F. Problem Formulation

In this paper, we aim at minimizing the maximum end-to-end latency of the uplink transmission among all devices, which we refer to as the system delay. We formulate this problem as follows:

$$\mathcal{P}1: \min_{\{o_k, g_k, \tau_k, f_k^c\}} \max_{k \in \mathcal{K}} t_k \tag{8a}$$

$$\text{s.t. } \text{SSIM}_k \geq \eta_k, \forall k \in \mathcal{K}, \tag{8b}$$

$$\sum_{k=1}^{K} \tau_k \leq 1, \tag{8c}$$

$$\sum_{k=1}^{K} f_k^c \leq F^c, \tag{8d}$$

$$g_k \geq 0, \forall k \in \mathcal{K}, \tag{8e}$$

$$o_k \in \{c_1, c_2, \ldots, c_N\}, \forall k \in \mathcal{K}, \tag{8f}$$

$$\tau_k \geq 0, f_k^c \geq 0, \forall k \in \mathcal{K}, \tag{8g}$$

where η_k denotes the performance requirement of each device, and (8b) ensures the SSIM requirement is met for each device. (8c) limits the overall communication resource of all devices, (8d) ensures the overall computation resource allocated to decode the message for each device can not exceed the threshold, (8e) is the truncation threshold constraint to ensure that it is a non-negative value, (8f) limits the range of the compression ratio within a given set. It can be observed that $\mathcal{P}1$ is a mixed integer non-linear problem (MINLP), which is hard to solve. In the next section, we will develop an effective algorithm to solve this problem.

III. Proposed Solution

In this section, we first transform the $\mathcal{P}1$ into an equivalent problem, then we propose an efficient method to address the transformed problem.

A. Problem Transformation

We reformulate the original problem into a more tractable one by introducing an auxiliary variable T as follows [13].

$$\mathcal{P}2: \min_{\{T,\tau_k,g_k,o_k,f_k^c\}} T \tag{9a}$$

$$s.t. \frac{L_k C_k^l}{f_k^l} + \frac{L_k D_0 o_k}{M e^{-g_k}} \frac{T_s}{\tau_k} + \frac{L_k C_k^d}{f_k^c} \leq T, \forall k \in \mathcal{K}, \tag{9b}$$

$$(8b) - (8g). \tag{9c}$$

Let $\{T^*, \tau_k^*, g_k^*, o_k^*, f_k^{c^*}\}$ be the optimal solution to the problem. We then obtain the following lemma.

Lemma 1: Solution to $\mathcal{P}2$ with $T < T^*$ is infeasible while the solution with $T > T^*$ is feasible.

Proof: The detailed proof is omitted here due to limited page. It can be found in extended version in https://github.com/cky-lab/proof/tree/main. ■

Based on the Lemma 1, we provide the algorithm to obtain the solution of $\mathcal{P}2$ by bisection search, as presented in Algorithm 1. Specifically, we now derive the upper bound and lower bound of T, i.e., T_{max} and T_{min}, to initialize the searching space. Intuitively, any feasible solution to the problem $\mathcal{P}2$ with a T' can be viewed as the upper bound, since the optimal delay is no more than this solution, i.e., $T^* \leq T'$. For instance, we can equally allocated communication and computation resource to all devices, i.e., $\tau_1 = ... = \tau_K = 1/K, f_1^c = ... = f_K^c = F^c/K$, and we can randomly choose available g_k, o_k to obtain the upper bound. Regarding the lower bound of T, we can assume that there is just one device k to be served and all the resources are allocated to the device, and g_k, o_k are optimized respectively.

It is not straight forward to test the feasibility of $\mathcal{P}2$ when T is fixed. Instead, we transform $\mathcal{P}2$ into an equivalent problem $\mathcal{P}3$ as follows. It is evident that the solution to $\mathcal{P}2$ is feasible if the objective function value of the $\mathcal{P}3$ is less than 1 to satisfy the communication resource allocation constraint in (8c).

$$\mathcal{P}3: \min_{\{\tau_k,g_k,o_k,f_k^c\}} \sum_{k\in\mathcal{K}} \tau_k \tag{10a}$$

$$s.t. \frac{L_k C_k^l}{f_k^l} + \frac{L_k D_0 o_k}{M e^{-g_k}} \frac{T_s}{\tau_k} + \frac{L_k C_k^d}{f_k^c} \leq T, \forall k \in \mathcal{K}, \tag{10b}$$

$$\text{SSIM}_k \geq \eta_k, \forall k \in \mathcal{K}, \tag{10c}$$

$$\sum_{k=1}^{K} f_k^c \leq F^c, \tag{10d}$$

$$g_k \geq 0, \forall k \in \mathcal{K}, \tag{10e}$$

$$o_k \in \{c_1, c_2, \ldots, c_N\}, \forall k \in \mathcal{K}, \tag{10f}$$

$$\tau_k \geq 0, f_k^c \geq 0, \forall k \in \mathcal{K}. \tag{10g}$$

Algorithm 1 The Optimal Algorithm to $\mathcal{P}1$

1: Initialize T_{min}, T_{max}, and the tolerance ε
2: **repeat**
3: Set $T = (T_{max} + T_{min})/2$.
4: Check the feasibility of the solution to $\mathcal{P}2$.
5: If $\{T, \tau_k', g_k', o_k', f_k^{c'}\}$ is a feasible solution to the problem, set $T_{max} = T$, else, set $T_{min} = T$.
6: **until** $(T_{max} - T_{min})/T_{max} \leq \varepsilon$.

B. Optimal Solution

In this subsection, we will discuss how to obtain the optimal solution to the $\mathcal{P}3$, which is a NP-hard problem. To make it more tractable, we first focus on a simple scenario where the compression ratio of each device is fixed. Accordingly, the problem can be formulated as

$$\mathcal{P}4: \min_{\{\tau_k,g_k,f_k^c\}} \sum_{k\in\mathcal{K}} \tau_k \tag{11a}$$

$$s.t. g_k \geq d_k, \forall k \in \mathcal{K}, \tag{11b}$$

$$(10b), (10d) - (10g), \tag{11c}$$

where d_k can be obtained by solving $\int_{d_k}^{+\infty} \frac{1}{g} e^{-g} dg = c_k$ using bisection method, and c_k is given as $c_k = \frac{P_k}{M r_k^\alpha \sigma^2} 10^{\frac{\ln\left(\frac{A_{o_k,2}-\eta_k}{\eta_k - A_{o_k,1}}\right) + C_{o_k,2}}{10 C_{o_k,1}}}$ by substituting (4) into (7). We have the following theorem.

Theorem 1: The problem $\mathcal{P}4$ is a convex optimization problem.

Proof: The objective function and constraints (10c) and (10d) are linear when o_k is fixed. We can prove the convexity of (10b) by computing its Hessian matrix and showing that the Hessian matrix is positive-definite. The detailed proof can be founded in the extended version. ■

Based on Theorem 1, we can utilize the Lagrangian method to solve $\mathcal{P}4$. The partial Lagrangian function to $\mathcal{P}4$ can be given by

$$\mathcal{L} = \sum_{k\in\mathcal{K}} \tau_k + \sum_{k\in\mathcal{K}} \lambda_k \left(\frac{L_k C_k^l}{f_k^l} + \frac{L_k D_0 o_k}{M e^{-g_k}} \frac{T_s}{\tau_k} + \frac{L_k C_k^d}{f_k^c} - T\right) + \mu\left(\sum_{k\in\mathcal{K}} f_k^c - F^c\right), \tag{12}$$

where λ_k and μ are the Lagrange multipliers associated with the constraints (10b) and (10d), respectively. Let $f_k^{c^*}$, τ_k^* and g_k^* denote the optimal solution to the $\mathcal{P}4$. Then, by utilizing the Karush-Kuhn-Tucker (KKT) conditions, we can derive the following theorem.

Theorem 2: The optimal solution to $\mathcal{P}4$ is given by

$$f_k^{c^*} = \frac{L_k}{T - \frac{L_k C_k^l}{f_k^l}} \left(\sqrt{\frac{D_0 o_k e^{g_k^*} T_s C_k^d}{M \mu^*}} + C_k^d\right), \forall k \in \mathcal{K}, \tag{13}$$

$$\tau_k^* = \frac{L_k}{T - \frac{L_k C_k^l}{f_k^l}} \left(\frac{D_0 o_k e^{g_k^*} T_s}{M} + \sqrt{\frac{\mu^* D_0 o_k e^{g_k^*} T_s C_k^d}{M}} \right), \quad (14)$$

$$g_k^* = d_k, \forall k \in \mathcal{K}, \quad (15)$$

where μ^* is the optimal Lagrange multiplier, as

$$\mu^* = \left(\frac{\sum_k \frac{L_k}{T - L_k C_k^l / f_k^l} \sqrt{\frac{D_0 o_k e^{g_k^*} T_s C_k^d}{M}}}{F^c - \sum_k \frac{L_k C_k^d}{T - L_k C_k^l / f_k^l}} \right)^2. \quad (16)$$

Proof: The proof can be found in extended version. ■

Remark 1*:* We can find that the optimal allocated computing resource for device k at BS , $f_k^{c^*}$, increases with the local computational latency $\frac{L_k C_k^l}{f_k^l}$ according to (13). Intuitively, if a device takes too much time on the local computation, the edge will allocate more computational resource to the device so as to reduce the end-to-end latency for the device such that to minimize the maximum latency among all devices.

So far, we have derived the optimal solution to $\mathcal{P}4$. Now we can obtain the algorithm to test the feasibility of problem $\mathcal{P}3$. One straight forward way is to exhaustively search the overall candidate compression ratio set $\{o_1, ..., o_K\}$ for devices and solve the corresponding $\mathcal{P}4$, then test whether the optimal objective value is smaller than 1. Thus we can obtain optimal solution to $\mathcal{P}1$ by this exhaustively search method. However, there are N^K possible values for $\{o_1, ..., o_K\}$, which exhibits exponential complexity. In the sequel, we will propose a heuristic algorithm with low complexity.

C. The Proposed Low-Complexity Algorithm

In this subsection, we propose a heuristic algorithm to address the $\mathcal{P}1$ with low complexity. Recall the optimal τ_k^* in (14), we note that the optimal τ_k^* is determined by $o_k e^{g_k}$. Moreover, since the objective of $\mathcal{P}3$ is the minimization of the sum of τ_k, we can choose the compression ratio with the smallest $o_k e^{g_k}$ for each device individually. The detailed information of the heuristic algorithm is presented in Algorithm 2. The computational complexity of the bisection search of T in the outer iteration is $\mathcal{O}(\log(1/\varepsilon))$. Meanwhile, the computational complexity of choosing the compression ratio for each device is $\mathcal{O}(KN \log(1/\varepsilon_2))$, where ε_2 is the error tolerance of the bisection method in computing d_k. Therefore, the overall computational complexity of the heuristic algorithm is $\mathcal{O}(\log(1/\varepsilon) KN \log(1/\varepsilon_2))$.

Algorithm 2 The heuristic algorithm for $\mathcal{P}1$

```
Initialize T_min, T_max, and the tolerance ε
repeat
    Set T = (T_max + T_min)/2.
    for Each device k do
        for o_k ∈ {c_1, c_2, ..., c_N} do
            Find the minimum o_k e^{g_k} that satisfy the performance constraint.
        end for
    end for
    Solve the problem P4 by Theorem 2, test whether the objective value is smaller than 1 or not. If true, set T_max = T, else, set T_min = T.
until (T_max − T_min)/T_max ≤ ε.
```

IV. Simulation Results

In this section, we present the simulations to demonstrate the performance of the proposed algorithm. The simulation parameters are set as follows unless otherwise stated. The BS has a coverage of 100 m, and we assume the path loss exponent $\alpha = 3$, the number of sub-channels M = 256, the bandwidth of each channel is 15 kHz, the noise variance is $\sigma^2 = -80$ dBm. All devices have the same transmit power 0.1 W. The number of images for each device to transmit is uniformly generated from 1 to 10. The SSIM requirement for each device follows the uniform distribution within $\eta_k \in [0.8, 0.93]$. We adopt the DeepJSCC model in [4] trained on the ImageNet [14] with a fixed SNR of 10 dB for training, and the SNR during testing varies from -10 to 20 dB. The size of the image is 128×128. The optional compression ratio set is $\{1/6, 1/8, 1/12, 1/24\}$. We assume that the local CPU frequency of each device is uniformly distributed in [1, 2] GHz. The edge server is equipped with Intel(R) Core(TM) i7-11700F with 16 cores with 4.9 GHz per core. We use the *cpulimit* [15] to control the CPU usage of a process (denoted in percentage). Through simulation, we obtain that the computation cost per pixel for the encoders and decoders is about 2170 CPU cycles/pixel and 2510 CPU cycles/pixel in average, respectively.

The optimal solution derived in Section III B is referred to as *OPT* and the heuristic algorithm proposed in Section III C is referred to as *HEU*. For comparison, we consider three benchmarks. The first one equally allocates the communication and computation resources to each decoder, while the compression ratio and truncation threshold for each device are then optimized, named as *EQU*. The second one equally allocates the computation and computation resources, and the compression ratios are fixed to the maximum for all devices while the truncation thresholds are optimized (ensure that the performance constraint can be satisfied), named as *FIX_O*. The third one also equally allocates the computation and computation resources, and the compression ratio is fixed to the maximum for each device, and the truncation threshold of each device is fixed to 0.5 (similarly, set the threshold to a high value to ensure that the performance constraint can be satisfied), named as *FIX_G*.

Fig. 3 shows the delay versus the number of devices. The computation resource at the edge server we use are two cores in total, which is denoted by 200%. It can be seen that the delay increases with the number of devices for all schemes. We note that both the optimal method and the heuristic algorithm always outperform the benchmarks. We can observe that heuristic algorithm achieves almost the same performance as the optimal solution. The *EQU* scheme shows performance

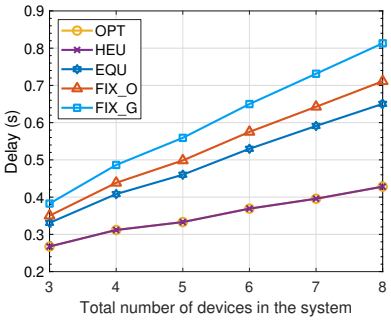

Fig. 3. Delay vs. number of devices.

degradation due to the fact that it can not utilize the communication and computation resource effectively. Besides, *FIX_O* performs even worse since it will transmit additional symbols when the performance constraint is low, which takes additional transmission time. Moreover, *FIX_G* degrades the performance even more because it deactivates too many channels when the performance constraint is low, which leads to a larger transmission delay.

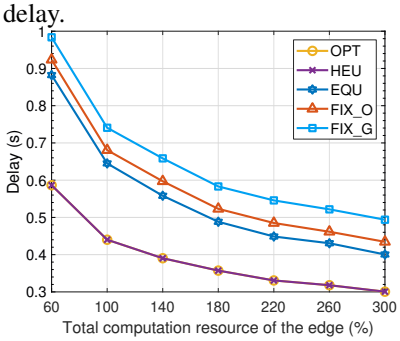

Fig. 4. Delay vs. edge computation resource. (The percentage means the percentage of a CPU core, for example, 300% means 3 CPU cores.)

Fig. 4 depicts the system delay versus the edge computation resource, where the number of devices is set to 5. From the figure, we observe that the system delay decreases with the edge computation resource for all methods. Besides, both the optimal algorithm and heuristic algorithm outperform the benchmarks.

Fig. 5 shows the relationship between local computation resource and edge computation resource. All devices transmit 5 images to the BS. The computation resource at the device 1 varies from 1 to 4 GHz, while the computation resource at device 2, device 3, device 4, device 5 are fixed as 1.5 GHz, 2 GHz, 2.5 GHz, 3 GHz, respectively. As shown in the figure, with the increase of local computation resource at device 1, the percentage of edge computation resource allocated to decode message of device 1 decreases, while the edge computation resource allocated to other devices increases. This is intuitive due to the fact that the local computation time of device 1 will decrease with increasing local computation resource, thus less resource should be allocated to device 1 at the edge to ensure the fairness among devices, which is aligned with the optimal edge resource allocation policy in (13).

V. Conclusion

In this paper, we proposed a resource allocation scheme to minimize the end-to-end latency of the uplink DL-based JSCC

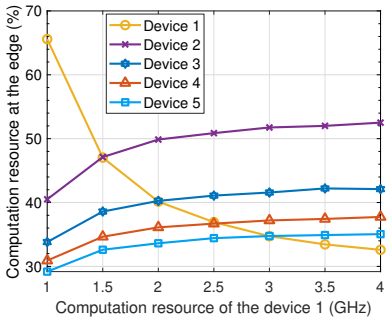

Fig. 5. Computation resource allocation vs. local computation resource of device 1. (200% in total, which means 2 CPU cores.)

systems. We analyzed the relationship between the end-to-end delay and the task performance of each device and then formulated the latency optimization problem, which is NP-hard. Through the problem transformation, we derived the closed form solution to the optimal compression ratio and channel truncation threshold selection policy and resource allocation strategy. Then we proposed an effective heuristic algorithm to solve the original problem with low computational complexity. Finally, simulation results demonstrated that both the proposed optimal algorithm and the heuristic algorithm can reduce end-to-end latency significantly. Remarkably, the proposed heuristic algorithm achieved nearly the same performance to the optimal solution but with much lower complexity.

References

[1] T. M. Cover and J. A. Thomas, "Information theory and statistics," *Elements Inf. Theory*, vol. 1, no. 1, pp. 279–335, 1991.

[2] S. Jiang, Z. Lin, Y. Li, Y. Shu, and Y. Liu, "Flexible High-resolution Object Detection on Edge Devices with Tunable Latency," *Proc. Annu. Int. Conf. Mob. Comput. Netw. (MobiCom)*, Oct. 2021, pp. 559–572.

[3] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep learning enabled semantic communication systems," *IEEE Trans. Signal Process.*, vol. 69, pp. 2663–2675, Apr. 2021.

[4] E. Bourtsoulatze, D. B. Kurka, and D. Gündüz, "Deep joint source-channel coding for wireless image transmission," *IEEE Trans. Cogn. Commun. Netw.*, vol. 5, no. 3, pp. 567–579, May 2019.

[5] T. Han, Q. Yang, Z. Shi, S. He, and Z. Zhang, "Semantic-preserved communication system for highly efficient speech transmission," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 1, pp. 245–259, Jan. 2023.

[6] Y. Wang, M. Chen, W. Saad, T. Luo, S. Cui, and H. V. Poor, "Performance optimization for semantic communications: An attention-based reinforcement learning approach," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 9, pp. 2598–2613, Sep. 2022.

[7] L. Yan, Z. Qin, R. Zhang, Y. Li, and G. Y. Li, "Resource allocation for text semantic communications," *IEEE Wireless Commun. Lett.*, pp. 1394–1398, Apr. 2022.

[8] X. Mu, Y. Liu, L. Guo, and N. Al-Dhahir, "Heterogeneous semantic and bit communications: A semi-NOMA scheme," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 1, pp. 155–169, Jan. 2023.

[9] Y. He, J. Ren, G. Yu, and Y. Cai, "Optimizing the learning performance in mobile augmented reality systems with CNN," *IEEE Trans. Wireless Commun.*, vol. 19, no. 8, pp. 5333–5344, Aug. 2020.

[10] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 1, pp. 491–506, Jan. 2020.

[11] H. Chen, F. Xing, Q. Yang, Y. Shu, Z. Shi, J. Chen, and Z. Tao, "A Lightweight Mobile-Anchor-based Multi-Target Outdoor Localization Scheme using LoRa Communication," *IEEE Trans. Green Commun. Netw.*, vol. 7, no. 4, pp. 1607–1619, Dec. 2023.

[12] K. Chi, Q. Yang, Z. Yang, Y. Duan, and Z. Zhang, "Resource allocation for capacity optimization in joint source-channel coding systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2023, pp.2099–2104.

[13] Z. Yang, M. Chen, W. Saad, C. S. Hong, and M. Shikh-Bahaei, "Delay minimization for federated learning over wireless communication networks," in *Proc. Int. Conf. Mach. Learn. Workshop*, Jul. 2020, pp. 1–7.

[14] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[15] A. Marletta. (2012). *Cpulimit*. [Online]. Available: https://github.com/opsengine/cpulimit.