Zhao, F., Bagwe, G., Mohammed, E., Feng, L., Zhang, L. and Sun, Y. (2023) Joint Computing Resource and Bandwidth Allocation for Semantic Communication Networks. In: 2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall), Hong Kong, 10-13 Oct 2023, ISBN 9798350329285 (doi: [10.1109/VTC2023-Fall60731.2023.10333793](https://doi.org/10.1109/VTC2023-Fall60731.2023.10333793))

Joint Computing Resource and Bandwidth Allocation for Semantic Communication Networks

Fangzhou Zhao*, Gaurav Bagwe†, Ezedin Mohammed†, Lei Feng‡, Lan Zhang† and Yao Sun*
*James Watt School of Engineering, University of Glasgow, Glasgow, UK
†Department of Electrical and Computer Engineering, Michigan Technological University, Michigan, USA
‡State Key Laboratory of Networking and Switching Technology, BUPT, Beijing, China
f.zhao.1@research.gla.ac.uk, Yao.Sun@glasgow.ac.uk

***Abstract*—As a new communication paradigm, neural network-driven semantic communication (SemCom) has demonstrated considerable promise in enhancing resource efficiency by transmitting the semantics rather than all bits of source information. Using a large semantic coding model can accurately distil semantics, and significantly save the required bandwidth. However, this consumes a large amount of computing resources, which are also precious in the network. In this paper, we investigate the joint computing resources and bandwidth allocation for SemCom networks. We first introduce the computing latency model in SemCom, and formulate the joint computing resources and bandwidth allocation optimization problem with the objective of maximizing semantic accuracy. Then, we transform this problem into a deep reinforcement learning framework and exploit a multi-agent proximal policy optimization to solve it. Numerical results show that the proposed method significantly improves the average semantic accuracy in the resource-constrained cases, compared with the two baselines.**

***Index Terms*—Semantic communication, bandwidth allocation, semantic coding model**

I. Introduction

With the explosive growth of new wireless applications, an unprecedented amount of data exerts strain on the capacity of wireless networks [1]. This motivates a new paradigm referred to as semantic communication (SemCom), with the goal of transmitting extracted semantics rather than all binary bits of source information [2], [3]. Specifically, SemCom employs a neural network-based semantic coding model to encode the source information into a few transmitted bits represented semantics and the semantics are decoded at the receiver end to reconstruct the source information [4]. By transmitting semantics to reduce network traffic, SemCom shows great potential to alleviate bandwidth limitations [5].

Utilizing a large-scale semantic coding model (SCM) enables the accurate distil of semantic information, resulting in transmitting fewer bits and consequently saving bandwidth [6]. Although SemCom offers substantial promise in saving bandwidth, neural network execution in SCM is a computing-intensive process that requires additional computing resources [7], [8]. This leads to a shortage of computing resources, resulting in high computing latency. Therefore, the exploration of joint computing resources and bandwidth allocation is meaningful in SemCom.

Regarding the joint computing resources and bandwidth allocation, there has been some research in traditional communication [9]. However, these researches cannot be directly extended to SemCom due to the inherent changes in the transmission scheme in SemCom, which poses two key challenges. First, different from traditional communication, the limitation of computing resources in SemCom is mainly reflected in the long computing latency of executing the SCM. This is because the execution of the neural network incurs an additional computing latency, which is significantly higher compared with that in traditional communication [10]. Meanwhile, when employing equivalent computing resources, the computing latency of an SCM increases with its neural network scale. Therefore, it is necessary to formulate a computing latency model of SemCom based on various SCMs. Second, the SemCom scheme introduces background knowledge and neural networks in order to transmit the meaning of the message, this makes it unique in the paradigm of data transmission. Thus, traditional resource allocation methods cannot directly be used in SemCom to joint computing resources and bandwidth allocation.

In this paper, we investigate the joint computing resources and bandwidth allocation for SemCom and propose a deep reinforcement learning (DRL) method to address it. The main contributions of this paper are summarized as follows.

- *The modeling of computing latency in SemCom:* The computing latency in SemCom consists of the inference latency of the neural network and the traditional transmission latency. Specially, we model the inference latency as the sum of the running time of all the kernels in the neural network and calculate the inference latency by estimating the running time of each kernel.
- *A DRL framework for joint computing resources and bandwidth allocation:* We formulate the joint resource allocation problem of computing resources and bandwidth in SemCom as a multi-agent Markov Decision Process (MDP) and define the actions, environment, and rewards accordingly.
- *MAPPO-based resource allocation algorithm:* We design a multi-agent proximal policy optimization (MAPPO) based algorithm to solve the formulated problem with the aim of maximizing the average semantic accuracy of users.
- *Simulation verification:* We conduct numerical simula-

tions to verify the effectiveness of the proposed method, where the results show the gain in terms of average semantic accuracy compared with the random method and the bandwidth-prior method.

The rest of this paper is organized as follows. Section II describes the SemCom system model. Section III introduces the proposed DRL method for joint computing resources and bandwidth allocation. Section IV presents simulation results and Section V concludes the paper.

II. System Model

We consider a SemCom-empowered cellular network consisting of a resource-constrained base station (BS) and a group of users (i.e., devices), as shown in Fig. 1. We assume that the semantic encoding and decoding of all transmissions take place on devices. Moreover, each device stores multiple SCMs with different computing consumption and varying semantic accuracy for specific tasks. In the considered process of end-to-end SemCom, the sender first selects an appropriate SCM for encoding and decoding. Subsequently, the source information is encoded to extract the semantics and packed into binary bits to be transmitted over the wireless channel. On the receiver side, the same SCM is exploited to recover the semantics based on the received bits and decode the meaning of the original information.

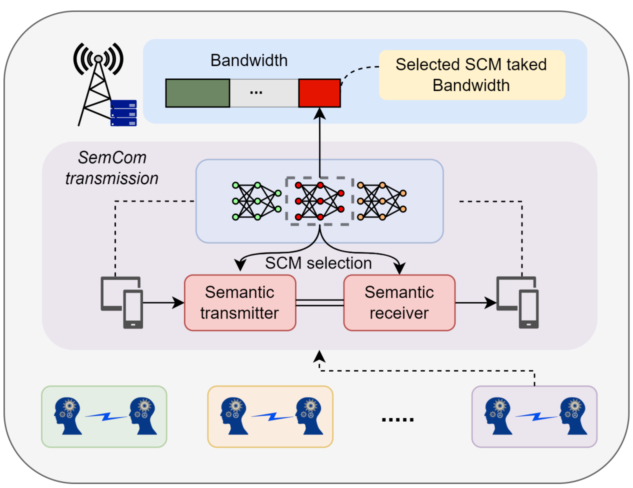

Fig. 1. The structure of semantic communication system

A. SemCom Computing Latency Model

We consider that each user $i \in \mathcal{U} = \{1, 2, 3, ..., N\}$ has one or multiple SCMs available for a specific communication task. We assume that for each SCM $j \in \mathcal{M} = \{1, 2, 3, ..., M\}$ in a user i there is a variable semantic compression rate $o_{i,j}(p)$, where p is the parameter of the SCM and determined by the length of the SCM output. Specifically, $o_{i,j}(p)$ is calculated as $o_{i,j}(p) = \frac{|s'_{i,j}|}{|s_{i,j}|}$ where $|s_{i,j}|$ is the number of bits of source information $s_{i,j}$ and $|s'_{i,j}|$ is the number of bits indicating semantics. For a given source information $s_{i,j}$, there is a minimum description length to represent it in bits which is denoted as $d(s_{i,j})$ [11]. Thus the range of the semantic compression rate is $o_{i,j}(p) \in [\delta_{min}, \delta_{max}]$, where δ_{max} is bound by $\delta_{max} \geq \left|\frac{d(s_{i,j})}{|s_{i,j}|}\right|$. Under a given channel, the transmission bandwidth $B_{i,j}$ required for SCM j with compression rate $o_{i,j}(p)$ is fixed.

Assuming user i has a required transmission rate ω_i, and the transmission latency is calculated as

$$\tau_{i,j}^t = \frac{s'_{i,j}}{\omega_i}. \tag{1}$$

Moreover, for user i running SCM j with a semantic compression rate $o_{i,j}(p)$, the computing latency $\tau_{i,j}^c$ is given by the neural network inference latency. Based on the device status, by predicting the running time of each kernel in the neural network the overall inference time can be calculated, and the predicting method can be found in [12]. For a kernel $z \in \mathcal{Z}_j$ where $\mathcal{Z}_j$ is the set of kernels in SCM j, denote $f(\cdot)$ as the kernel prediction function, the computing latency can be calculated as

$$\tau_{i,j}^c = \sum_{z \in \mathcal{Z}_j} f(z) + \epsilon_z, \tag{2}$$

where ϵ_z is the prediction error of z. Thus the overall latency caused by transmission and computing is

$$\tau_{i,j} = \tau_{i,j}^t + \tau_{i,j}^c. \tag{3}$$

B. Joint Computing Resources and Bandwidth Allocation Problem Formulation

The optimization objective of the joint computing resources and bandwidth allocation problem is to maximize the average semantic accuracy of all users by selecting SCM and semantic compression rate. Moreover, we consider the bandwidth constraint, latency constraint, and the accuracy of semantic constraints. Let B be the total bandwidth in the considered BS. For each user i, the maximum latency is denoted as τ_i^{max}. Meanwhile, assuming that each user i has a minimum semantic accuracy requirement E_i^{min} to ensure transmission quality. The optimization problem can be described as

$$\max_{j, o_{i,j}(p)} \quad \hat{E} = \sum_{i=1}^{n} \frac{E(j, o_{i,j}(p))}{n} \tag{4}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} B_{i,j} \leq B, i \in \mathcal{U}, j \in \mathcal{M}, \tag{4-1}$$

$$\tau_i \leq \tau_i^{max}, i \in \mathcal{U}, j \in \mathcal{M}, \tag{4-2}$$

$$E(j, o_{i,j}(p)) \geq E_i^{min}, i \in \mathcal{U}, j \in \mathcal{M}, \tag{4-3}$$

$$o_{i,j}(p) \in [\delta_{min}, \delta_{max}], i \in \mathcal{U}, j \in \mathcal{M}.. \tag{4-4}$$

where $E(j, o_{i,j}(p))$ is the semantic accuracy that the user i using SCM j with semantic compression rate $o_{i,j}(p)$. In particular, we should note that $E(j, o_{i,j}(p))$ is the output of the neural network, and it cannot be precisely encapsulated by a rigorous mathematical model. In this case, to optimize $\hat{E}$, we transform the above optimization problem into the DRL framework to solve this problem.

III. DRL-Based Joint Computing Resource and Bandwidth Allocation

In this section, we first transform the joint computing resources and bandwidth allocation problem into a DRL framework and then exploit MAPPO to solve this DRL problem.

A. DRL Framework

Based on the system model described in Section II, the joint computing resources and bandwidth allocation can be formulated as follows multi-agent MDP problem. To illustrate this problem let us define action, state, and reward.

1) *Action:* In this system, each user acts as an agent to select suitable SCMs for SemCom transmission. We denote A_i as the set for the action space of selecting SCM and compression rate for user i at time t. Specifically, an action $a_{i,t} \in A_i$ is a decision matrix with length $n \times m$.
2) *State:* The state of the system at time t is denoted by $s_t = \{\mathcal{C}_t, \mathcal{D}_t\}$, where $\mathcal{C}_t = \{C_{t,1}, C_{t,2}, ..., C_{t,n}\}$ is the set of channel condition (i.e. SNR) and $\mathcal{D}_t = \{D_{t,1}, D_{t,2}, ..., D_{t,n}\}$ is available computing resources of each user. For given channel condition $C_{t,i}$ and available computing resources $D_{t,i}$, when selecting SCM j and compression rate $o_{i,j}(p)$, B_i and $\tau^c_{i,j}$ can be determined.
3) *Reward:* To reflect the performance of the action, we denote the immediate reward in time t as

$$r_t = \sum_{k=1}^{n} E_t(j, o_{i,j}(p)), \tag{5}$$

where $E_t(j, o_{i,j}(p))$ is the semantic accuracy that the user i using SCM j with semantic compression rate $o_{i,j}(p)$ in time t. The aim of the DRL is to find an optimal policy that achieves the maximum long-term reward from the state. Thus, we define the long-term reward as

$$R = \sum_{i=0}^{\infty} \gamma_i r_{t+i}, \tag{6}$$

where $\gamma_i \in [0, 1]$ is the discount factor, which indicates the extent to which future rewards and immediate rewards affect overall returns.

B. DRL-based Model Training

In this paper, we use the centralized training and decentralized execution method to implement reinforcement learning and use the MAPPO algorithm to solve the target problem [13]. In this section, we first describe the training method for a single agent and then introduce the global training process for centralized training and decentralized execution.

The key to reinforcement learning is to execute actions by estimating the policy function and the value function. The agent selects the action according to the policy function, then the value function gives the action reward, this process repeats until finish training. Let us introduce the training methods of

Algorithm 1: MAPPO resource allocation algorathm

Initialize the value network q and policy networks π, and the parameter θ, w;
Set learning rate α;
while *training not stop* **do**
  for *u=1,2,..,n* **do**
    Select an action $a_{i,t}$ base on the gradient or select random action;
    Input the bandwidth consumption B_i, semantic compression$o_{i,j}(p)$, required transmission rate$|s'_{i,j}|$, predicted running time $\tau^c_{i,j}$;
    Calculate the total latency $\tau_{i,j}$;
    Returns semantic accuracy $E(j, o_{i,j}(p))$;
  end
  Calculate the overall bandwidth consumption average semantic accuracy $\hat{E}$;
  Observe the reward r_t and new state s_{t+1};
  for *u=1,2,..,n* **do**
    update R;
    Computing the target loss function according to (10) and the gradient descent step is performed on (12), thus updating θ, w;
  end
end

the policy function and the value function. For user i at time t, the state-value function can be expressed as

$$V_\pi(s_t) = \sum\nolimits_a \pi(a_{i,t}|s_t) \cdot Q_\pi(s_t, a_{i,t}), \tag{7}$$

where V_π is state-value function when taking action $a_{i,t}$ in the current state s_t. We approximate the state-value function by $V(s_t, \theta, w)$ which denoted by

$$V(s_t, \theta, w) = \sum\nolimits_a \pi(a_{i,t}|s_t, \theta) \cdot q(s_t, a_{i,t}, w), \tag{8}$$

where $V(s_t, \theta, w)$ is the neural network approximation state-value function, the policy function and the value function are approximated by the policy network $\pi(a_{i,t}|s_t, \theta)$ and the value network $q(s_t, a_{i,t}, w)$ respectively, θ is the parameter of the policy network and w is the parameter of the value network.

We centralized training the value network $q(s_t, a_{i,t}, w)$ which is trained on the base station with all agent's actions and observations, and all agents use the same value network. The value network is updated by the TD algorithm and gradient descent. Specifically, we use a stochastic gradient descent (SGD) scheme to randomly select one of the optional SCMs to calculate the TD error, and then calculate the stochastic gradient. TD target is denoted as

$$y_t = r_t + \gamma Q(s_{t+1}, a_{i,t+1}, w_t), \tag{9}$$

then loss function can be expressed as

$$L_t(w) = \frac{1}{2}[Q(s_t, a_{i,t}, w) - y_t]^2, \tag{10}$$

and the parameters w can be updated according to the following rules:

$$w_{t+1} = w_t - \alpha \cdot \frac{\partial L_t(w)}{\partial w}|_{w=w_t}, \quad (11)$$

where α denotes the learning rate of the value network.

Once the value network training has been completed, local policy network training is performed. In decentralized execution, we train the policy network $\pi(a_{i,t}|s_t, \theta)$ locally for each user, with actions $a_{i,t}$ and state s_t as their independent decisions and observations. The Monte Carlo estimation function of the policy gradient can be expressed as

$$g(a_{i,t}, \theta) = \frac{\partial \log \pi(a_{i,t}|s_t, \theta)}{\partial \theta} \cdot q(s_t, a_{i,t}, w), \quad (12)$$

then the parameters θ can be updated according to the gradient ascent:

$$\theta_{t+1} = \theta_t + \beta \cdot g(a_{i,t}, \theta), \quad (13)$$

where β is the learning rate of the policy network.

By continuously updating the policy network and the value network, agents can select actions with higher rewards. The MAPPO-based joint computing resource and bandwidth allocation algorithm is shown in Algorithm 1.

IV. Numerical Results and Discussions

We conduct numerical simulations to validate the semantic accuracy of the proposed method. We compare two baselines that execute two other action policies in the formulated DRL framework:

1) Bandwidth-prior policy: The actions in this baseline are always selecting the lowest bandwidth consumption SCM and semantic compression rate;
2) Random policy: The actions in this baseline are randomly selecting SCM and semantic compression rate.

A. Simulation Settings

TABLE I
SIMULATION PARAMETERS

Parameter	Value
Number of user n	50
SCM of each user	1 to 10
Total bandwidth	100Mhz
Semantic accuracy range	[0.85, 0.99]
Running time of each SCM	20ms to 1000ms
latency capacity of each user	600ms to 1200ms

We consider a SemCom scenario consisting of a BS and 50 user devices, and each user device is randomly set up with 1 to 10 well-trained SCM models. For each SCM we assume that there is a fixed bandwidth requirement and computing latency. We set the BS to have 100Mhz bandwidth. In addition, the running time of each SCM is randomly generated within 20ms to 1000ms. For each user, the semantic accuracy of each SCM is in the range of [0.85, 0.99]. In particular, it is assumed that the semantic compression rate of each SCM is variable, and in the simulation, we set three different semantic compression rates: original compression rate, compressed to 150% of the original compression rate, and compressed to 50% of the original compression rate.

B. Results and Discussions

We first examine the convergence of the MAPPO-based joint computing resource and bandwidth allocation algorithm network. Fig.3 shows the successful convergence of the MAPPO method in the given numerical environment after 350,000 steps of training, which shows the effectiveness of the trained network.

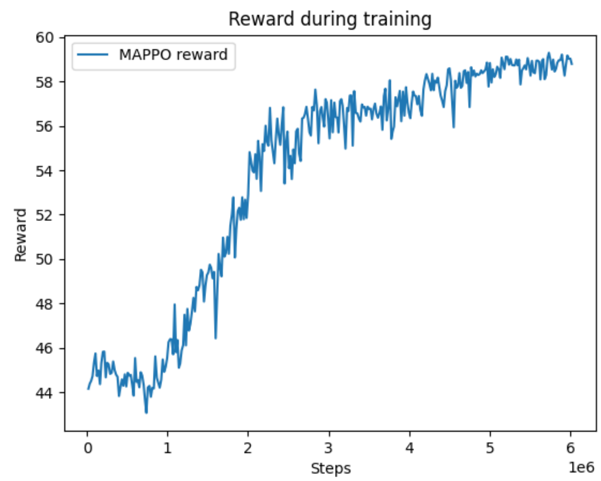

Fig. 2. Convergence of MAPPO-based joint computing resource and bandwidth allocation algorithm network

Then, we evaluate the average rewards after network convergence. The comparison of the average reward of our method with other baselines in the 200-step iterations performed after the network converges is shown in TableII. The rewards of the MAPPO policy finally converge at 58.850, which is significantly better than the bandwidth-prior policy and the random policy. The reason is, that the bandwidth-prior policy always selects the action with the worst semantic accuracy and smallest bandwidth consumption and thus has the worst reward. The random strategy can select some other actions and achives a better average reward than the bandwidth-prior policy. Our method relies on the MAPPO algorithm to make a reasonable decision from a global perspective so as to obtain the highest average reward.

TABLE II
FINAL AVERAGE REWARD AFTER NETWORK CONVERGENCE

Policy	Average reward
MAPPO policy	58.850
Bandwidth-prior policy	42,843
Random policy	14.682

To evaluate the stability of the proposed method, we compare the rewards with training steps after network convergence

Fig. 3. Rewards versus steps after network convergence

in Fig.3. The reward gained by the MAPPO policy is more fluctuating but overall significantly higher than the other two methods. This arises from the extensive scope of the formulated environmental and action spaces. In contrast to the other two baselines, our method has a broader strategic scape, consequently leading to higher fluctuation. Our method in the worst case also is better than the other two baselines, this is due to the centralized training considering the global awards, avoiding the most extreme resource competition between agents.

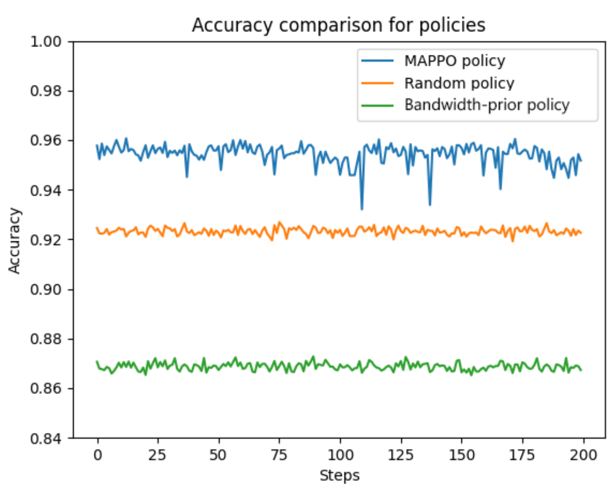

Fig. 4. Semantic accuracy versus steps after network convergence

At last, we present the average semantic accuracy with training steps after network convergence as shown in Fig.4. Similar to the rewards, the average semantic accuracy of the MAPPO policy fluctuates significantly but is still significantly higher than that of the other two baselines. Our method selects actions with higher rewards from a global perspective and the defined reward function directly reflects the average semantic accuracy, thus exhibiting the highest average semantic accuracy as expected.

V. Conclusion

In this paper, we investigate the problem of joint computing resources and bandwidth allocation for semantic communication (SemCom). Specifically, we formulate the optimization problem and transform it into a reinforcement learning framework to solve it. Compared with the two baselines, the numerical results presented that our method achieves higher average semantic accuracy. Our work can be seen as a pioneer in the management of multi-dimensional resources in SemCom. The initial results can be extended to some sophisticated models, and later guide network operators to better design protocols for SemCom deployment.

Acknowledgment

This work of Lan Zhang, Guarav Bagwe, and Ezedin Mohammed, was supported by the National Science Foundation under Grants CCF-2106754, CCF-2221741, CCF-2153381, and CCF-2151238.

References

[1] E. C. Strinati and S. Barbarossa, "6G networks: Beyond Shannon towards semantic and goal-oriented communications," *Computer Networks*, vol. 190, p. 107930, 2021.

[2] J. Bao, P. Basu, M. Dean, C. Partridge, A. Swami, W. Leland, and J. A. Hendler, "Towards a theory of semantic communication," in *2011 IEEE Network Science Workshop*. IEEE, 2011, pp. 110–117.

[3] L. Xia, Y. Sun, D. Niyato, D. Feng, L. Feng, and M. A. Imran, "xURLLC-Aware Service Provisioning in Vehicular Networks: A Semantic Communication Perspective," *arXiv preprint arXiv:2302.11993*, 2023.

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

[5] G. Shi, Y. Xiao, Y. Li, and X. Xie, "From semantic communication to semantic-aware networking: Model, architecture, and open problems," *IEEE Communications Magazine*, vol. 59, no. 8, pp. 44–50, 2021.

[6] F. Zhao, Y. Sun, R. Cheng, and M. A. Imran, "Background knowledge aware semantic coding model selection," in *2022 IEEE 22nd International Conference on Communication Technology (ICCT)*, 2022, pp. 84–89.

[7] Z. Qin, X. Tao, J. Lu, W. Tong, and G. Y. Li, "Semantic communications: Principles and challenges," *arXiv preprint arXiv:2201.01389*, 2021.

[8] L. Xia, Y. Sun, D. Niyato, X. Li, and M. A. Imran, "Joint user association and bandwidth allocation in semantic communication networks," *arXiv preprint arXiv:2106.00574*, 2023.

[9] S. Li, S. Lin, L. Cai, W. Li, and G. Zhu, "Joint Resource Allocation and Computation Offloading With Time-Varying Fading Channel in Vehicular Edge Computing," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3384–3398, 2020.

[10] W. Yang, H. Du, Z. Q. Liew, W. Y. B. Lim, Z. Xiong, D. Niyato, X. Chi, X. S. Shen, and C. Miao, "Semantic communications for future internet: Fundamentals, applications, and challenges," *IEEE Communications Surveys & Tutorials*, 2022.

[11] K. Niu, J. Dai, S. Yao, S. Wang, Z. Si, X. Qin, and P. Zhang, "A paradigm shift toward semantic communications," *IEEE Communications Magazine*, vol. 60, no. 11, pp. 113–119, 2022.

[12] L. L. Zhang, S. Han, J. Wei, N. Zheng, T. Cao, Y. Yang, and Y. Liu, "Nn-meter: Towards accurate latency prediction of deep-learning model inference on diverse edge devices," in *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 81–93. [Online]. Available: https://doi.org/10.1145/3458864.3467882

[13] C. Yu, A. Velu, E. Vinitsky, J. Gao, Y. Wang, A. Bayen, and Y. Wu, "The surprising effectiveness of ppo in cooperative multi-agent games," *Advances in Neural Information Processing Systems*, vol. 35, pp. 24 611–24 624, 2022.