Self-Critical Alternate Learning-Based Semantic Broadcast Communication

Zhilin Lu, Rongpeng Li, *Senior Member, IEEE*, Ming Lei, *Member, IEEE*, Chan Wang, Zhifeng Zhao, *Member, IEEE*, and Honggang Zhang, *Fellow, IEEE*

***Abstract*— Semantic communication (SemCom) has been deemed as a promising communication paradigm to break through the bottleneck of traditional communications. Nonetheless, most of the existing works focus more on point-to-point communication scenarios and its extension to multi-user scenarios is not that straightforward due to its cost-inefficiencies to directly scale the joint source-channel coding (JSCC) framework to the multi-user communication system. Meanwhile, previous methods optimize the system by differentiable bit-level supervision, easily leading to a "semantic gap". Therefore, we delve into multi-user broadcast communication (BC) based on the universal transformer (UT) and propose a reinforcement learning (RL) based self-critical alternate learning (SCAL) algorithm, named SemanticBC-SCAL, to capably adapt to the different BC channels from one transmitter (TX) to multiple receivers (RXs) for sentence generation task. In particular, to enable stable optimization via a non-differentiable semantic metric, we regard sentence similarity as a reward and formulate this learning process as an RL problem. Considering the huge decision space, we adopt a lightweight but efficient self-critical supervision to guide the learning process. Meanwhile, an alternate learning mechanism is developed to provide cost-effective learning, in which the encoder and decoders are updated asynchronously as independent agents. Notably, the incorporation of RL makes SemanticBC-SCAL compliant with any user-defined semantic similarity metric and simultaneously addresses the channel non-differentiability issue by alternate learning. Besides, the convergence of SemanticBC-SCAL is also theoretically established. Extensive simulation results have been conducted to verify the effectiveness and superiorness of our approach, especially in low signal-to-noise ratio regions.**

***Index Terms*— Semantic broadcast communication, self-critical reinforcement learning, non-differentiable channel, multi-user communications, semantic similarity.**

I. Introduction

With the rapid development of communication technologies and the rise of artificial intelligence (AI), tremendous intelligent applications, such as extended reality (XR), holographic communication and autonomous driving, flourish and impose stringent communication requirements [1], which are beyond the capabilities of 5G. In order to deal with this performance dilemma, researchers are resorting to reap the merits of AI, and deep learning (DL) enabled semantic communication (SemCom) emerges as one of the promising solutions. In particular, targeted at the semantic and/or effectiveness level communications [2], SemCom has attained intense research interest [3] with encouraging results in text transmission [4], [5], [6], [7], [8], speech/audio transmission [9], [10], [11], and image/video transmission [12], [13], [14], [15], [16].

However, it is noteworthy that most of the literature predominantly focuses on point-to-point communication scenarios, while shedding little light on exploring one-to-many semantic broadcast communication (BC). For example, [17] proposes to estimate the signal-to-noise ratio (SNR) at the decoder before adaptively decoding transmitted images. Reference [18] extends the unidirectional and bidirectional decoders to a degraded broadcast channel, and RXs therein decode the text in different languages yet with the same meanings as needed. Similarly, [19] designs a semantic recognizer DistilBERT to distinguish different receivers according to emotional properties (i.e., positive or negative). Different from [17], [18], and [19] that broadcast all extracted features, the semantic encoder in [20] extracts disentangled semantic features and only broadcasts the receiver's intended semantic information to further improve transmission efficiency. Apparently, these efforts on semantic BC lay the very foundation for resource-efficient transmission to multiple receivers requiring the same content [21].

Besides, existing semantic BC schemes [17], [18], [19], [20], [21] continuously adopt JSCC to optimize the end-to-end (E2E) deep neural networks (DNNs) encompassing multiple receivers. Hence, it is cost-ineffective to directly scale the JSCC framework to multi-user communication systems (e.g., semantic BC), due to the awful size of parameters corresponding to exponentially increased receivers. Instead, an alternative

Received 3 December 2023; revised 24 May 2024 and 29 August 2024; accepted 14 October 2024. Date of publication 28 October 2024; date of current version 19 May 2025. This work was supported in part by the National Key Research and Development Program of China under Grant 2024YFE0200600, in part by the National Natural Science Foundation of China under Grant 62071425, in part by the Zhejiang Key Research and Development Plan under Grant 2022C01093, in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LR23F010005, in part by the National Key Laboratory of Wireless Communications Foundation under Grant 2023KP01601, and in part by the Big Data and Intelligent Computing Key Lab of CQUPT under Grant BDIC-2023-B-001. The associate editor coordinating the review of this article and approving it for publication was H. Kim. *(Corresponding author: Rongpeng Li.)*

Zhilin Lu, Rongpeng Li, Ming Lei, and Chan Wang are with Zhejiang University, Hangzhou 310027, China (e-mail: lu_zhilin@zju.edu.cn; lirongpeng@zju.edu.cn; lm1029@zju.edu.cn; 0617464@zju.edu.cn).

Zhifeng Zhao is with Zhejiang Laboratory, Hangzhou 311121, China, and also with Zhejiang University, Hangzhou 310027, China (e-mail: zhaozf@zhejianglab.com).

Honggang Zhang is with City University of Macau, Macau 999078, China (e-mail: hgzhang@cityu.edu.mo).

Digital Object Identifier 10.1109/TCOMM.2024.3487513

framework, which is built on top of a state-of-the-art transformer DNN structure (e.g., universal transformer [UT] [22]) and capably separates the training of TX and RXs, is highly essential. In this regard, alternate learning between TX and RXs sounds intuitive, but its convergence could be questionable. Furthermore, in existing works, the objective function commonly leverages a differentiable function, such as cross-entropy (CE) or mean square error (MSE), so as to facilitate direct gradient backpropagation. This kind of bit-level metric, despite being feasible in most cases, inevitably introduces a "semantic gap" [3] due to inconsistencies between evaluation metrics and optimization objectives, i.e., the training stage is optimized by the differentiable CE or MSE function, while the test stage is evaluated by non-differentiable semantic metrics, such as BLEU, CIDEr [3]. Meanwhile, the difficulty of dealing with the non-differentiability of semantic similarity metrics could be exaggerated by the large dimension of sequence decoding space, which demands a simple algorithm. In addition, most of these SemCom approaches impractically assume the channel is known and differentiable to enable direct backpropagation through the channel. It remains challenging to solve the non-differentiability issue in SemCom.

In summary, the design of a semantic BC with one TX and multiple RXs is still hindered by the following critical issues:

Q1: (Scalability) How to design a cost-effective semantic BC framework?

Q2: (Compatibility) How to compatibly deal with the non-differentiability of semantic similarity metrics as well as practical channel propagation, so as to develop a learning-efficient algorithm?

Q3: (Convergence) How to establish a theoretical guidance for alternate learning and calibrate the convergent hyperparameters correspondingly?

In this paper, we jointly address these critical yet challenging issues and propose a convergent semantic BC framework based on self-critical alternate learning (SCAL), named SemanticBC-SCAL. In particular, SemanticBC-SCAL employs the UT [22] as the backbone to more efficiently adjust the number of processing steps according to the complexity of each token. Furthermore, SemanticBC-SCAL capably adapts to different channel environments from one TX to multiple RXs and competently solves the non-differentiability issue in the objective function (i.e., semantic similarity metrics) by incorporating RL techniques that regard the semantic similarity as the reward rather than computing the gradient. Meanwhile, considering the difficulty of handling large semantic decision space, SemanticBC-SCAL utilizes a lightweight self-critical supervision [23], which capably provides a simple and expedited baseline estimation with low variance, to yield a stable and efficient gradient policy from pre-trained candidates. Furthermore, SemanticBC-SCAL is equipped with an asynchronous alternate learning mechanism to unify TX & RXs, swiftly adapting to different numbers of RXs, and also addresses the non-differentiable channel. The primary contributions of our work are summarized as follows:

- We propose a semantic BC framework, named SemanticBC-SCAL, comprising one TX and multiple RXs, built upon the universal transformer (UT) architecture [22], the superiority of which has been widely validated in [24] and [6]. Notably, the decoders in SemanticBC-SCAL share a consistent structure with trained parameters, which facilitates further adaptive learning. Thus, it allows the trained decoder model to readily expand to additional RXs at a reasonable cost, effectively addressing the *Q1*.
- Within this proposed SemanticBC-SCAL framework, in order to enable stable optimization via a non-differentiable semantic similarity metric, we regard semantic similarity as a reward and formulate it as an RL problem, which is compliant with any semantic metric [8]. Meanwhile, we adopt self-critical supervision to provide simple and efficient gradient estimation. This becomes especially suitable for complex SemCom systems with large semantic decision space and partially addresses the mentioned *Q2*.
- Combining the above approaches, a self-critical based alternate learning mechanism is devised to alternately train the encoder at TX and multiple decoders at RXs, as the conventional supervised training can not directly perform backpropagation through non-differentiable channels. In particular, under self-critical supervision, when the TX remains fixed, multiple RXs locally update their parameters according to their unique channel properties. After certain iterations, the RXs are fixed, and the TX updates itself by aggregating the information of multiple RXs uniformly. This alternate learning mechanism addresses the non-differentiable channel in *Q2*. Moreover, we carefully investigate the theoretical convergence and validate the performance. The results show that the RXs can adapt to different channels, and yield superior performance compared to traditional communication method [25] and JSCC-based semantic model [5]. This addresses the mentioned *Q3*.

The rest of this paper is mainly organized as follows. Section II briefly explains the recent works and clarifies the novelty of our work. In Section III, we introduce the preliminaries of the semantic BC model and formulate the optimization problem. In Section IV, we describe the alternate learning mechanism and self-critical supervision of the SemanticBC-SCAL framework and analyze the theoretical convergence. In Section V, the numerical results are illustrated and discussed. Finally, conclusions are drawn in Section VI.

II. Related Works

SemCom primarily concentrates on the source information reconstruction based on the received information. For text-based SemCom systems, Farsad et al. [4] propose the initial JSCC based on the recurrent neural network (RNN) codecs for fixed length sentences, which shows that the DL-based codecs are capable of achieving lower word error rate and preserve semantic information by embedding sentences into a semantic space. Inspired by this success, Xie et al. [5] design a transformer-based SemCom system, named DeepSC, which aims to recover the semantics for

TABLE I
SUMMARY AND COMPARISON OF SEMANTIC BC SCHEMES

Refs	Brief description	Limitations
[17]	It is an SNR-adaptive deep JSCC scheme to minimize image distortion for multiple users.	The objective function is bit level MSE function that lacks semantic level supervision and the JSCC-based optimization framework is not easy to scale to more users.
[18]	It extends the proposed unidirectional and bidirectional decoders to the degraded broadcast channel.	The broadcast case is only designed for two users and lacks scalability.
[19]	It leverages users' semantic features to receive the desired information.	The adaptability of different channels has not been discussed and analyzed.
[20]	It is a feature-disentangled semantic BC framework.	The theoretical convergence has not been further analyzed.

arbitrary lengths of sentences under varying channels. Since then, SemCom has attracted a lot of attention and many works utilize JSCC frameworks to transmit sentence semantics [6], [7], [26], such as UT [22] based semantic coding system [6], [24], and mutual information (MI) based performance optimization between semantic compression and semantic fidelity. Moreover, Wang et al. [27] and Seo et al. [28] adopt knowledge graph (KG) and probability distribution respectively to infer semantics information explicitly. In addition to text, Weng et al. [9] design an attention-based E2E SemCom system for speech transmission. Tong [10] et al. develop a federated learning (FL) trained model to deliver audio semantics between multiple devices and the server. Besides, Bourtsoulatze et al. [12] initially extend the JSCC framework and apply it to the deep image SemCom system, which provides graceful degradation as SNR changes. Correspondingly, Kurka et al. in [13] consider the channel feedback to enhance the image reconstruction quality based on the JSCC scheme. Moreover, the authors in [15] and [16] introduce the semantics and transmit some key motion points to maximize overall visual quality, thus improving transmission efficiency.

Apart from the achievements in semantic level SemCom, some results are achieved at the effectiveness level as well, by transmitting essential semantics timely for successful task accomplishment. For example, Jankowski et al. [29], [30] target the identification of individuals or vehicles and design two alternative (digital and analog communications-based) schemes to enhance retrieval accuracy. Kountouris et al. [31] develop a communication paradigm, in which smart devices sample the "semantics of information" to steer their traffic for real-time remote actuation. Additionally, in [32], a transformer-based framework is designed to unify the DNN structure of the transmitter for different tasks with different types of data (i.e., single-modal data and multi-modal data), wherein many-to-many and many-to-one scenarios are exemplified. Similarly, [33] also presents a unified E2E framework that can serve different tasks with multiple modalities by dynamically adjusting feature dimensions and DNN layers. Commonly, existing works in SemCom focus more on point-to-point or multiple-encoder-single-decoder JSCC frameworks. Furthermore, as explained earlier, there emerge several works [17], [18], [19], [20] on Semantic BC, and the related drawbacks are summarized in Table I. Notably, these semantic BC works tend to encounter challenges related to scalability, compatibility, and convergence issues.

In addition, there exist some efforts to combine RL and SemCom to improve the communication efficiency of multi-agent systems [27], [34], [35], [36], [37]. For example, in order to alleviate information bottleneck (IB) and improve the effectiveness of the multi-round communication process, [34] applies an attention-based sampling mechanism to the graph convolution network for multi-agent cooperation. Different from the aforementioned works that utilize SemCom for effective transmission, [37] considers an RL-based semantic bit allocation to implement task-driven semantic coding for a traditional hybrid coding framework. Specifically, they design semantic maps for different tasks to extract the pixel-wise semantic fidelity and leverage RL to integrate the semantic fidelity metric into the in-loop optimization of semantic coding. It can be seen that RL is capable of improving communication efficiency and has demonstrated considerable advantages. Nevertheless, there are relatively few works using RL to develop an in-depth optimization for the SemCom system, despite its potential effectiveness in addressing scalability and compatibility issues.

III. SYSTEM MODEL AND PROBLEM FORMULATION

In this section, we introduce the basic UT [22] based semantic BC model and also give a brief description of related semantic metrics for performance evaluation. Beforehand, we summarize mainly used notations in Table II.

A. System Model

As illustrated in Fig. 1, the proposed BC system mainly includes three parts, that is, one TX, N random channels, and N RXs. Generally speaking, at TX side, the message $\boldsymbol{m}$ is first converted to an embedding by the embedding layer. Then the embedding is sent to implement semantic encoding, including a semantic encoder for feature extraction and a dense layer for channel encoding. Then, the quantization layer quantifies $\boldsymbol{x}$ as $\boldsymbol{b}$ to facilitate the subsequent transmission and this extracted valuable information is broadcast through different physical channels to reach RX_n ($n \sim N$).[1] Likewise, at RX_n side, the de-quantization layer detects and reconstructs the received symbol $\hat{\boldsymbol{b}}_n$ from the received noisy signals, and the semantic decoding, i.e., including channel decoder and semantic decoder, estimate and recover the source as $\hat{\boldsymbol{m}}_n$ based on the knowledge base (KB). Different from the

[1]The $\sim$ operator implies $n \in \{1, 2, \cdots N\}$.

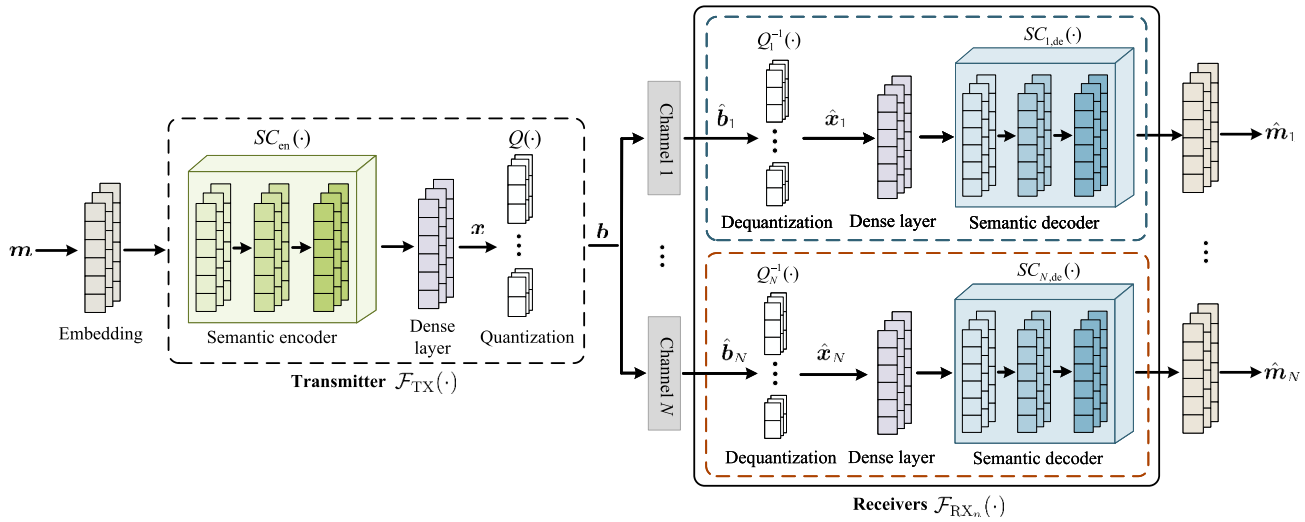


Fig. 1. The system model for semantic BC system.

TABLE II
NOTIONS USED IN THIS PAPER

Notation	Definition
N	The number of receivers
$\mathcal{F}_{\mathrm{TX}}, \mathcal{F}_{\mathrm{RX}_n}$	The abbreviated terms for transmitter and receiver n ($n \sim N$) respectively
$\boldsymbol{m}, \hat{\boldsymbol{m}}_n$	Input message of TX and decoded message of RX_n
Θ	Semantic similarity metric
$\theta_{\mathrm{en}}, \theta_{\mathrm{q}}$	Parameters for semantic encoder and quantization layer
$\phi_{n,\mathrm{de}}, \phi_{n,\mathrm{dq}}$	Parameters for semantic decoder and dequantization layer
$SC_{\mathrm{en}}(\cdot), SC_{n,\mathrm{de}}(\cdot)$	Semantic encoder and semantic decoder n respectively
$SC_{\mathrm{TX}}(\cdot), SC_{n,\mathrm{RX}}(\cdot)$	Semantic coding for TX, and semantic decoding for RX_n
$Q(\cdot), Q_n^{-1}(\cdot)$	Quantization and dequantization layer respectively
θ, ϕ_n	Parameters for TX and RX_n respectively
$T, \hat{T}_n$	The length of input message for TX and output message for RX_n respectively
D	Dimension of message embedding
$\mathcal{H}_n(\cdot)$	Random channel
W_m	The dictionary
M	Dimension of the dictionary
$p_n(\cdot)$	Probability of choosing a single word of semantic decoder n
K	Number of parallel samples
$\hat{\boldsymbol{m}}_{n,i}$	The i-th complete decoding trajectory for RX_n
$r_n^{(t)}$	Reward at step t for semantic decoder n
$s_n^{(t)}$	State at step t for semantic decoder n
$s_{n,\mathrm{de}}^{(t)}$	State at step t for semantic decoder n
$\mathcal{S}, \mathcal{A}$	The state space and action space
P	State transition probability function
R	Reward space
γ	Discounted factor
κ	Decoders' local iterations in each update cycle

classic SemCom [5], we adopt the UT [22] based semantic encoder/decoders to achieve semantic coding/decoding. Besides, RXs share the same structure but have individual DNN parameters accommodating to channel properties, so as to recover the original information as accurately as possible.

Mathematically, for a source message $\boldsymbol{m} = \{w^{(1)}, \cdots, w^{(i)}, \cdots, w^{(T)}\}$, where T is the sequence length, and $w^{(i)}$ can be regarded as a word from dictionary W_m (i.e., $w^{(i)} \in W_m$). The encoding process $\mathcal{F}_{\mathrm{TX}}(\cdot)$ (parameterized by θ) for TX consists of semantic encoding $SC_{\mathrm{en}}(\cdot)$ and quantization $Q(\cdot)$. The semantic encoding is given by

$$\boldsymbol{x} = SC_{\mathrm{en}}(\boldsymbol{m}; \theta_{\mathrm{en}}), \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^{T \times D}$, D is the dimension of each symbol in $\boldsymbol{x}$, $SC_{\mathrm{en}}(\cdot)$ is the combination of semantic encoder and channel encoder parameterized by θ_{en}.

After that, the embedding $\boldsymbol{x}$ is quantified by a θ_{q}-induced quantization $Q(\cdot)$ as

$$\boldsymbol{b} = Q(\boldsymbol{x}; \theta_{\mathrm{q}}), \tag{2}$$

where $\boldsymbol{b} \in \mathbb{R}^{T \times B}$, B is the number of bits for each quantified symbol of $\boldsymbol{x}$. Then TX broadcasts $\boldsymbol{b}$ to multiple physical channels.

At the RX_n side, different from the point-to-point transmission with a single receiver, the transmitted data in semantic BC experience heterogeneous channel conditions corresponding to different receivers. Analogously, the decoding process $\mathcal{F}_{\mathrm{RX}_n}(\cdot)$ (parameterized by ϕ_n) for RX_n involves the reverse steps, including dequantization and semantic decoding, which are represented as $Q_n^{-1}(\cdot)$ and $SC_{n,\mathrm{de}}(\cdot)$ respectively. With the aid of channel state information (CSI), the semantic decoder can be dynamically adjusted to achieve adaptive decoding. As such, the decoding message $\hat{\boldsymbol{m}}_n$ of RX_n can be denoted as

$$\hat{\boldsymbol{m}}_n = SC_{n,\mathrm{de}}\left(Q_n^{-1}(\mathcal{H}_n(\boldsymbol{b}); \phi_{n,\mathrm{dq}}); \phi_{n,\mathrm{de}}\right), \tag{3}$$

where $\mathcal{H}_n(\cdot)$ ($n \sim N$) is the channel layer and is usually modeled as additive white Gaussian noise (AWGN) channel or multiplicative Rayleigh fading channel [12].

As such, RX_n recovers the source message by predicting the probability distribution of the next word in the dictionary. In the end, a complete decoded message can be denoted as $\hat{\boldsymbol{m}}_n = \{\hat{w}_n^{(1)}, \hat{w}_n^{(2)}, \cdots, \hat{w}_n^{(\hat{T}_n)}\}$, where $\hat{T}_n$ is the length of decoded message for RX_n.

B. Problem Formulation

Given the described system model, our goal is to maximize semantic similarity for RX_n by learning the optimal parameters for the system $\langle \mathcal{F}_{\mathrm{TX}}, \mathcal{F}_{\mathrm{RX}_n} \rangle$, that is,

$$\langle \theta^*, \phi_n^* \rangle = \underset{\mathcal{F}_{\mathrm{TX}}, \mathcal{F}_{\mathrm{RX}_n}}{\arg\max}\ \Theta(\boldsymbol{m}, \hat{\boldsymbol{m}}_n), \tag{4}$$

where $\Theta(\cdot)$ can be any reasonable semantic similarity metric, and we do not assume its differentiability. Therefore, in this paper, bilingual evaluation understudy (BLEU) scores [38], BERT (bidirectional encoder representation from transformers) [39] based similarity metric (i.e., BERT-SIM) and word accuracy rate (WAR) are adopted to measure the degree of consistency between the input $\boldsymbol{m}$ and output $\hat{\boldsymbol{m}}_n$ [3], [5].

Nevertheless, the problem defined in (4) for semantic BC is challenging to solve by traditional JSCC frameworks. The difficulties are twofold. On the one hand, to address the inherent "semantic gap", the commonly used differentiable objective will no longer be applicable as the gradient cannot be directly propagated backward through the channel. On the other hand, along with the number of RXs increases, the scalability issue emerges, primarily as the commonly used point-to-point optimization method becomes invalid and untenable. Furthermore, to alleviate the "semantic gap", we formulate (4) as an RL problem by regarding non-differentiable semantic similarity as a reward. In addition, we adopt an alternate learning mechanism to cope with the scalability issue by iteratively updating the encoder and decoders, in which the encoder and decoders are all conceptualized as independent agents.

Alternate learning mechanism: Specially, upon freezing the parameters of $\mathcal{F}_{\mathrm{TX}}$ (i.e., θ), $\mathcal{F}_{\mathrm{RX}_n}$ is locally updated κ iterations. In contrast, when $\mathcal{F}_{\mathrm{RX}_n}$ (i.e., ϕ_n) is constant, $\mathcal{F}_{\mathrm{TX}}$ is updated once independently by averaging the decoders' results. It is worth emphasizing that the encoder updates once, while each decoder locally updates κ iterations, collectively forming a single *update cycle*. Corresponding to this alternate learning mechanism, the objective functions for the TX and RX_n diverge intrinsically. In this regard, the overall optimization objective outlined in (4) is formulated into the following two parts

$$\theta^* = \underset{\mathcal{F}_{\mathrm{TX}}(\cdot)}{\arg\max}\, \Theta(\boldsymbol{m}, \mathrm{avg}(\underbrace{\mathcal{F}_{\mathrm{RX}_n}(}_{\text{no grad}} \mathcal{H}_n(\mathcal{F}_{\mathrm{TX}}(\boldsymbol{m}))))); \tag{5a}$$

$$\phi_n^* = \underset{\mathcal{F}_{\mathrm{RX}_n}(\cdot)}{\arg\max}\, \Theta(\boldsymbol{m}, \mathcal{F}_{\mathrm{RX}_n}(\mathcal{H}_n(\underbrace{\mathcal{F}_{\mathrm{TX}}(\boldsymbol{m})}_{\text{no grad}}))), \tag{5b}$$

where $\mathrm{avg}(\cdot)$ is to get the mean value.

In this paper, we talk about how to alternately optimize (5a) and (5b) on top of RL, the details of which shall be presented in the next section.

IV. SemanticBC-SCAL Scheme With Alternating Learning Mechanism

In this section, we first elaborate on how to formulate a Markov decision process (MDP) to facilitate the application of RL, optimizing the semantic BC system with semantic-level supervision. Afterward, we talk about the self-critical alternate learning-based algorithm, SemanticBC-SCAL, to produce a solution. Furthermore, we also investigate its convergence.

A. The Markov Decision Process (MDP) Framework

Following this multi-user semantic BC system, where the encoder is formulated as one agent at TX while the decoders are modeled as N independent agents at RXs, which interacts with the environment (i.e. the text dictionary W_m) by taking actions on the basis of states and receiving the rewards from the environment.

Without loss of generality, the MDP setting can be denoted as $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ is the state space and the action space respectively, P is the state transition probability, and R denotes reward function with discounted factor $\gamma \in (0, 1]$. Since actions (i.e., word generation processes) are implemented at the decoders, the state space and action space are determined by RX_n and shared for the encoder. As such, we define the state space as $\mathcal{S} = \{s_1, \cdots, s_n, \cdots, s_N\}, n \sim N$ and action space (i.e., vocabulary) as $\mathcal{A} = \{w_1, \cdots, w_n, \cdots, w_N\}, w_n \in W_m$, where the dimension of dictionary W_m is M. Moreover, the state transition function is denoted as $P = \{P_{\pi_\theta}, P_{\pi_{\phi_n}}\}$, while the reward for TX and RX_n are distinctly parameterized by π_θ and π_{ϕ_n} respectively, which are further delineated as $R = \{R_{\pi_\theta}, R_{\pi_{\phi_n}}\}$.

In particular, the MDP can be detailed as follows.

- *Action*. The definition of action is to generate the next token $a_n^{(t)} = \hat{w}_n^{(t)}$ from the dictionary W_m at RX_n, where the sequence generation process starts with token "<SOS>" and ends with "<EOS>". In contrast to traditional communication where there exists a one-to-one relationship with the source message $\boldsymbol{m}$, SemCom allows the existence of multiple candidate interpretations. Hence, the state-action space can be larger than traditional methods.
- *State*. We define the state $s_n^{(t)}$ as a combination of the sequential state $s_{n,\mathrm{de}}^{(t)}$ of decoder n at step t and decoded tokens (e.g., generated words for the text task) until step t, i.e., $s_n^{(t)} = \{s_{n,\mathrm{de}}^{(t)}, \hat{w}_n^{(1)}, \hat{w}_n^{(2)}, \cdots, \hat{w}_n^{(t)}\} \in \mathcal{S}$. It merits emphasis that our goal is to interpret messages at the semantic level, which may lead to variations in length between the decoded sentences and their original counterparts. Here, for different RX_n, the maximum length of the decoded sentence is $\hat{T}_n$. Meanwhile, once the next token $\hat{w}_n^{(t+1)}$ is generated, the state transition is deterministic between any adjacent state.
- *Policy*. Notably, the policies for TX and RXs shall be independently calibrated, since the semantic encoder at TX aims to optimize the encoding process in (5a), while the semantic decoder at RX_n is responsible for generating semantically accurate messages by maximizing (5b).

Specifically, for TX side, the semantic encoding of the encoder is continuous and only needs to be implemented once for each sample. In this sense, we model this process as a continuous Gaussian distribution with the mean value $\boldsymbol{\mu} = \mathcal{F}_{\mathrm{TX}}(\boldsymbol{m}) \in \mathbb{R}^{1\times D}$, and the covariance matrix $\Sigma = (\sigma \boldsymbol{I})^2 \in \mathbb{R}^{D\times D}$ (where σ is typically set to 0.1) [8]. In other words, some Gaussian noise is intentionally added to the semantic encoding process, to encourage a certain exploration. Mathematically, the encoder policy can be written as

$$\pi_\theta := \text{Sample}\left(\mathcal{N}\left(\boldsymbol{\mu} = \mathcal{F}_{\mathrm{TX}}(\boldsymbol{m}), \Sigma = (\sigma \boldsymbol{I})^2\right)\right), \quad (6)$$

where $\mathcal{N}$ indicates a Gaussian distribution.

For RX_n, instead of one deterministic distribution, the input of the decoder is decoded by K times and the output likelihood is modeled as a probabilistic multinomial distribution (termed as $\mathcal{MD}$) under self-critical supervision (it will be detailed in Section IV-B), which means RX_n can obtain a group of different output from K views, so as to better reap the potentially diversified semantics of the same sentence. As such, sampling from this kind of multinomial distribution ensures the collection of samples with the maximum likelihood from multi-view. For each multinomial distribution, it is modeled by the "softmax" function [8] that is parameterized on the ϕ_n. Then for step t, one has

$$\pi_{\phi_n}^{(t)} := \text{Sample}(\mathcal{MD}([p_n(\hat{w}_1^{(t)}), p_n(\hat{w}_2^{(t)}), \cdots, p_n(\hat{w}_M^{(t)})])). \quad (7)$$

- *State transition probability.* According to the abovementioned definitions, the state transition function P is deterministic between two adjacent states. Formally, we have $P(s_n^{(t+1)}|s_n^{(t)}, \hat{w}_n^{(t)}) = 1$.
- *Reward.* We use the semantic similarity as a reward and denoted as $\Theta(\boldsymbol{m}, \hat{\boldsymbol{m}}_n)$. In the sentence generation model, the reward can only be obtained until the last token is generated. Therefore, the reward is sparse in SemanticBC-SCAL. In other words, the intermediate reward is always zero except for the last time step $\hat{T}_n$, that is,

$$r_n^{(t)} = \begin{cases} 0, & \text{if } t \neq \hat{T}_n; \\ \Theta(\boldsymbol{m}, \hat{\boldsymbol{m}}_n), & \text{if } t = \hat{T}_n. \end{cases} \quad (8)$$

For the sake of simplification, we assume the discounted factor $\gamma = 1$, so without loss of generality, the *return* for our system is formulated as

$$G_n^{(t)} = \sum_{k=t+1}^{\hat{T}_n} \gamma^{k-t-1} r_n^{(k)} = \sum_{t=1}^{\hat{T}_n} r_n^{(t)}. \quad (9)$$

Accordingly, the state-value function is given by $V(s_n^{(t)}) = \mathbb{E}\left[G_n^{(t)}|s_n^{(t)}\right]$. Consistent with the reformulated problem in (5), at decoder sides, the objective function for RX_n is to maximize the value function, which can be viewed as maximizing *return* for a complete trajectory $\{\hat{w}_n^{(t)}, s_n^{(t+1)}, \hat{w}_n^{(t+1)}, \cdots\}$ under decoder policy π_{ϕ_n} starting from $s_n^{(0)}$, which is given by

$$J(\phi_n) = V_{\pi_{\phi_n}}(s_n^{(0)}) = \mathbb{E}_{\pi_{\phi_n}}\left[G_n^0|s_n^0\right] = \mathbb{E}_{\pi_{\phi_n}}\left[\sum_{t=1}^{\hat{T}_n} r_n^{(t)}\right]. \quad (10)$$

While for encoder optimization, TX aims to find an optimal encoding to maximize the semantic accuracy for all decoders. Under the encoder policy π_θ, we regard the maximum average *return* of all decoders as the objective function, that is,

$$J(\theta) = \frac{1}{N}\sum_{n=1}^{N} V_{\pi_\theta}(s_n^{(0)}) = \frac{1}{N}\sum_{n=1}^{N} \mathbb{E}_{\pi_\theta}\left[\sum_{t=1}^{\hat{T}_n} r_n^{(t)}\right]. \quad (11)$$

B. Self-Critical Optimization Under Alternate Learning Mechanism

In order to find a proper solution to maximize the objectives functions (10) and (11), one most straightforward solution is to simulate the environment to obtain Monte-Carlo trajectory $\{(s_n^{(t)}, \hat{w}_n^{(t)}, r_n^{(t)})\}_{t=1}^{\hat{T}_n}$ for each time-step t [40]. Though Monte-Carlo rollout provides an unbiased estimation for expected *return*, large state-action space unavoidably leads to high variance [8], [41]. Another alternative solution is the "actor-critic" method and its variants [42], in which the baseline is established by constructing another *value network* to criticize the policy [8]. However, this way will introduce extra parameters and estimation bias. Meanwhile, due to the extensive number of tokens included in the dictionary W_m, the size of state-action space exceeds 10^4 [8]. In this sense, merely the Monte-Carlo trajectory or actor-critic methods are impractical to implement for sentence generation in the SemCom system.

Based on the aforementioned discussions, to handle this large-scale sentence generation, we develop the SemanticBC-SCAL algorithm to combine a simper and practical RL algorithm, i.e., self-critical learning [8], [23], with the abovementioned alternate learning mechanism in Section III-B. Specially, SCAL samples K parallel Monte Carlo trajectories and computes the mean reward from the rest of $K-1$ parallel samples, that is, $\hat{\boldsymbol{m}}_n = \{\hat{\boldsymbol{m}}_{n,1}, \cdots, \hat{\boldsymbol{m}}_{n,K-1}\}$, to act as the baseline function. To generate a complete trajectory by introducing a certain exploration, the semantic encoder/decoders could employ the policy π_θ and π_{ϕ_n} respectively to guide the system in accommodating multi-view semantic representations. Since the baseline comes from SCAL itself, i.e., the other $K-1$ decoding results, this kind of supervision can be regarded as "self-critical" manner [23], [40], [43], and the empirical results show that larger K will have a larger absolute value of the mean and introduce lower variance [40]. Taking an example of the decoder optimization in Fig. 2, the TX sends the message ("I have just brought a banana") to the broadcast channel, and the RX_n first samples $K = 5$ parallel samples before channel decoding, in which the decoded parallel sentences

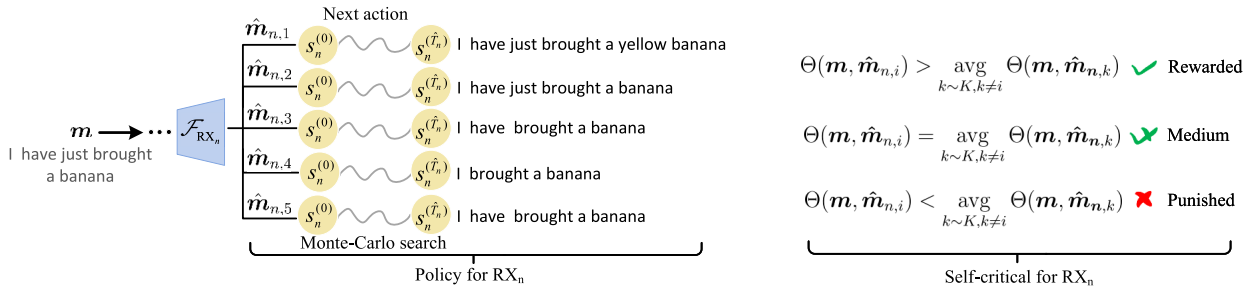


Fig. 2. Illustration of the self-critic optimization for decoder side at RX_n.

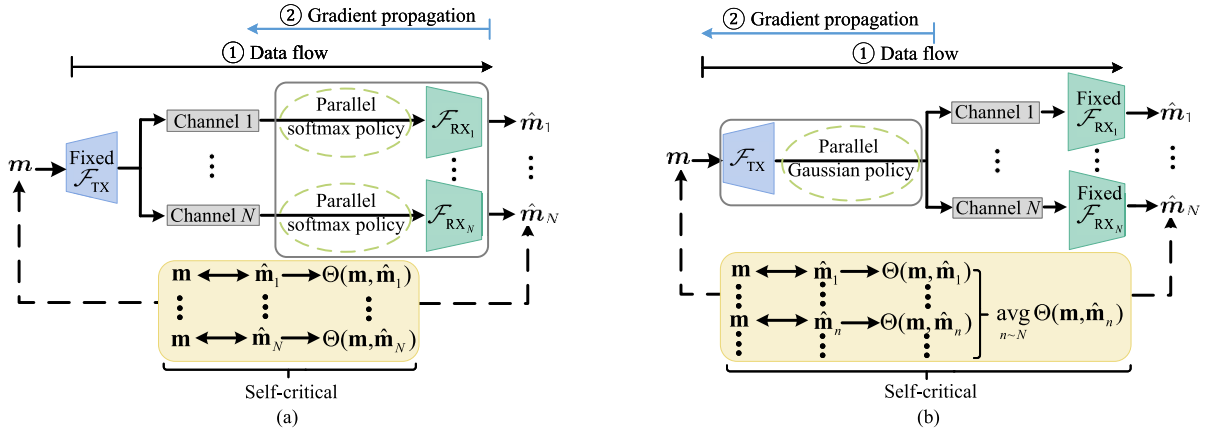


Fig. 3. Optimization process under the alternate learning mechanism. (a) The training process of RX_n. (b) The training process of TX. (The black solid line is the direction of data flow, while the blue solid line denotes the direction of gradient propagation.)

could be different. Then, RX_n can be viewed as an actor responsible for taking actions by Monte Carlo sampling to generate 5 parallel decoding results. As such, the advantage of one decoding result will be evaluated by the other 4 parallel samples.

On top of self-critical learning, an alternate learning mechanism is devised to asynchronously update the TX and multiple RXs to adapt to its channel environment. To be specific, we start with the decoder training stage, by freezing the encoder parameters θ and locally updating the decoder parameters ϕ_n. The details of the training process of RX_n in SemanticBC-SCAL are illustrated in Fig. 3(a). Based on the MDP framework and self-critical optimization with alternate learning, it is ready to analyze the semantic policy gradient for both RX_n and TX. Notably, we primarily consider the relative preference of one trajectory over others, which makes the gradient of (10) slightly different from those in classical RL approaches. Meanwhile, since the semantic similarity score can only be obtained via the self-critical manner after decoding a complete sentence, the decoder updating occurs in an episodic manner (i.e., only after a complete sentence transmission). After adopting certain well-calibrated approximations, Theorem 1 unveils the related result.

Theorem 1: (Self-critical semantic policy gradient for $\mathrm{RX_n}$*) For any semantic decoder* n *optimized by the alternate learning mechanism with* K *parallel sampling, suppose that the decoder is updated by the policy* $\pi_{\phi_n}^{(t)}$ *to generate a complete sequence* $\{\hat{w}_n^{(1)}, \hat{w}_n^{(2)}, \cdots, \hat{w}_n^{(\hat{T}_n)}\}$*, the gradient can be approximated by*

$$\nabla_{\phi_n} J(\phi_n) \approx \frac{1}{K} \sum\nolimits_{i=1}^{K} \left[\sum\nolimits_{t=1}^{\hat{T}_n} \nabla_{\phi_n} \log \pi_{\phi_{n,i}}(\hat{w}_n^{(t)} | s_n^{(t)}) \cdot \left(\Theta_{n,i} - \underset{k \sim K, k \neq i}{\mathrm{avg}} \Theta_{n,k} \right) \right], \tag{12}$$

where the subscript i *in* $\pi_{\phi_n,i}$ *denotes the* i*-th parallel sampling policy, while* $\mathrm{avg}(\cdot)$ *gets the mean value.* $\Theta_{n,k}$ *is the return of the sampling trajectory* i *for* RX_n*.*

Proof:

During the decoder optimization process, a complete trajectory $\hat{\boldsymbol{m}}_n = \{\hat{w}_n^{(1)}, \hat{w}_n^{(2)}, \cdots, \hat{w}_n^{(\hat{T}_n)}\}$ has the probability of $P(\hat{\boldsymbol{m}}_n | \phi_n)$, and the gradient of $\mathrm{RX_n}$ is

$$\begin{aligned} & J(\phi_n) \\ &= \nabla_{\phi_n} \left[\sum\nolimits_{\hat{\boldsymbol{m}}_n} P(\hat{\boldsymbol{m}}_n | \phi_n) \sum\nolimits_{t=1}^{\hat{T}_n} r_n^{(t)} \right] \\ &= \sum\nolimits_{\hat{\boldsymbol{m}}_n} \left[\nabla_{\phi_n} P(\hat{\boldsymbol{m}}_n | \phi_n) \cdot \sum\nolimits_{t=1}^{\hat{T}_n} r_n^{(t)} \right] \\ &\overset{(a)}{=} \sum\nolimits_{\hat{\boldsymbol{m}}_n} \left[P(\hat{\boldsymbol{m}}_n | \phi_n) \nabla_{\phi_n} \log \left(P(\hat{\boldsymbol{m}}_n | \phi_n) \right) \cdot \sum\nolimits_{t=1}^{\hat{T}_n} r_n^{(t)} \right] \\ &= \mathbb{E}_{\hat{\boldsymbol{m}}_n} \left[\nabla_{\phi_n} \log \left(P(\hat{\boldsymbol{m}}_n | \phi_n) \right) \cdot \sum\nolimits_{t=1}^{\hat{T}_n} r_n^{(t)} \right], \end{aligned} \tag{13}$$

where the equality (a) is derived by using the log-trick $\nabla \log x = \nabla x / x$. Since we sample one Monte Carlo trajectory

to estimate the $\hat{\boldsymbol{m}}_n$, then one obtains

$$\nabla_{\phi_n} J(\phi_n) \approx \nabla_{\phi_n} \log\left(P(\hat{\boldsymbol{m}}_n|\phi_n)\right) \cdot \sum\nolimits_{t=1}^{\hat{T}_n} r_n^{(t)}. \tag{14}$$

As the sentence is generated step by step through $\hat{T}_n+1$ actions, $P(\hat{\boldsymbol{m}}_n|\phi_n)$ is further written as $P(\hat{\boldsymbol{m}}_n|\phi_n) = p(s_n^0)\prod_{t=1}^{\hat{T}_n}\left[\pi_{\phi_n}(\hat{w}_n^{(t)}|s_n^{(t)})P_{\pi_{\phi_n}}(s_n^{(t)}|s_n^{(t-1)},\hat{w}_n^{(t-1)})\right]$. Meanwhile, the reward function is sparse, i.e., $\sum_{t=1}^{\hat{T}_n} r_n^{(t)} = \Theta(\boldsymbol{m},\hat{\boldsymbol{m}}_n)$, and the state transition probability $P_{\pi_{\phi_n}}(s_n^{(t)}|s_n^{(t-1)},\hat{w}_n^{(t-1)})$ is deterministic between two adjacent states. Therefore, one has

$$\nabla_{\phi_n} J(\theta,\phi_n) \approx \sum\nolimits_{t=1}^{\hat{T}_n} \nabla_{\phi_n} \log \pi_{\phi_n}(\hat{w}_n^{(t)}|s_n^{(t)})\Theta(\boldsymbol{m},\hat{\boldsymbol{m}}_n). \tag{15}$$

Finally, the self-critical baseline function is implemented by utilizing the mean reward from the rest of $K-1$ parallel samples, thus, the proof ends with (12). □

Remark: It is worth mentioning that the advantage of policy $\pi_{\phi_n,i}$ (i.e., $\left(\Theta_{n,i} - \underset{k\sim K,k\neq i}{\text{avg}}\ \Theta_{n,k}\right)$) manifests the differences with parallel learned samples and leads to a graceful gradient variance reduction [23]. Furthermore, in semantics representation, the one with higher semantic similarity gives a larger positive reward, while those with lower advantage will be punished. In other words, the semantic metric Θ_n of RX_n is closely linked to the gradient optimization direction, steering parameter updates toward maximizing the semantic metrics. Therefore, this approach not only improves learning efficiency but also ensures the generation of high-quality outputs that are closely aligned with the target performance.

As for the encoder, recalling that as in Fig. 3(b), the encoder output is converted to K parallel samples by Gaussian policy, which are then sent to broadcast channels. Therefore, as the objective of the encoder is to maximize the average reward of all decoders, the average semantic similarity in different decoders is conversely used to optimize the $\mathcal{F}_{\mathrm{TX}}$. In this way, the gradient takes effect for the policy only, while the semantic similarity performance, which implicitly embeds channel state information, is leveraged. Analogously, we give the optimization direction of (11) by Theorem 2.

Theorem 2: (Self-critical semantic policy gradient for TX) For semantic encoder optimized by the alternate learning mechanism, suppose the encoder is updated by the parallel Gaussian strategy π_θ, the gradient can be approximated by the Monte-Carlo sampling as

$$\nabla_\theta J(\theta) \approx \frac{1}{N}\cdot\frac{1}{K}\sum\nolimits_{n=1}^{N}\sum\nolimits_{i=1}^{K}\left[\left[\widetilde{\mathcal{F}_{\mathrm{TX}}}(\mathbf{m})-\mathcal{F}_{\mathrm{TX}}(\mathbf{m})\right]^T \times\boldsymbol{\Sigma}^{-1}\left[\nabla_\theta\mathcal{F}_{\mathrm{TX}}(\mathbf{m})\right]\cdot\left(\Theta_{n,i}-\underset{k\sim K,k\neq i}{\text{avg}}\ \Theta_{n,k}\right)\right], \tag{16}$$

where $\widetilde{\mathcal{F}_{\mathrm{TX}}}$ is the Gaussian sampling for $\mathbf{m}$, T is the length of source message $\boldsymbol{m}$.

Proof: The proof is similar to that for Theorem 1. Thus, we only provide a sketch here.

Algorithm 1 The Training Process of SemanticBC-SCAL

Input: Batch size, learning rate lr, parallel samples K, input sequence $\boldsymbol{m}$, local iterations κ, no. of pre-training end epochs E_p, no. of end epochs E_e, no. of end batches E_b, number of RXs N.

Output: Encoder parameter θ, decoder parameter ϕ_n.

// Pre-training stage

for epoch=1 : E_p **do**
  Sample a batch of data to train parameters of encoder θ and decoder ϕ optimized by CE loss [6].
  Update θ and ϕ to obtain the pre-trained parameters θ_{pre} and ϕ_{pre}.
end for

// RL-based alternate learning

$\theta \leftarrow \theta_{\text{pre}}$, $\phi_n \leftarrow \phi_{\text{pre}}$
for epoch=$E_p+1 : E_e$ **do**
  for $i = 1 : E_b$ **do**
    Sample a batch of data
    // Training Decoder
    while $(i \mod \kappa) \neq 0$ **do**
      for $n = 1 : N$ **do**
        Freeze θ, RX_n samples K parallel trajectories, and updates $\phi_n \leftarrow \phi_n + \mathrm{lr}\cdot\nabla_{\phi_n}\mathrm{J}(\phi_n)$ as in (12).
      end for
    end while
    // Training Encoder
    For each RX_n, freeze ϕ_n and sample K parallel trajectories.
    Update $\theta \leftarrow \theta + \mathrm{lr}\cdot\nabla_\theta \mathrm{J}(\theta)$ by (16).
  end for
end for

// Finish training

Return θ and ϕ_n.

Basically, for a Gaussian distribution

$$\pi_\theta(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{D/2}\sqrt{\det\boldsymbol{\Sigma}}}\exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right), \tag{17}$$

as the covariance matrix $\Sigma = (\sigma\boldsymbol{I})^2 \in \mathbb{R}^{D\times D}$ and σ is a pre-defined constant, we have

$$\nabla_\theta \log\pi_\theta(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\Sigma}) = \left[\boldsymbol{x}-\boldsymbol{\mu}\right]^T\boldsymbol{\Sigma}^{-1}\left[\nabla_\phi\boldsymbol{\mu}\right]. \tag{18}$$

Taking $\boldsymbol{x} = \widetilde{\mathcal{F}_{\mathrm{TX}}}$ and $\boldsymbol{\mu} = \mathcal{F}_{\mathrm{TX}}(\boldsymbol{m})$, as well as following the proof for Theorem 1, we have the theorem. □

Remark: On the basis of this stable and reasonable learning policy, our SCAL-based semantic BC system can adaptively encode/decode by introducing as few parameters and costs as possible. To obtain a stable and expected reward (semantic similarity score) in the training process, we adopt the aforementioned alternate learning mechanism at both TX and $\mathrm{RX_n}$ to implement a Monte-Carlo simulation. Specifically, in an *update cycle*, the RXs update κ iterations and then TX updates once. Finally, during this self-critical learning-based alternate training, we obtain a stable and precise gradient surrogate,

without introducing an additional module to estimate the gradient or suffering from a poor estimation of the expected reward.

Finally, the detailed procedures for training SemanticBC-SCAL are summarized in **Algorithm 1**.

C. Convergence Analysis

In this section, we analyze the convergence property of SemanticBC-SCAL. Beforehand, to ensure the convergence of the encoder and decoders, we first present some essential assumptions.

Assumption 1: The learning rates for encoder and decoder n are unified as $\mathrm{lr}_{\mathrm{en}} = \mathrm{lr}_{\mathrm{de}} = \mathrm{lr}(\alpha)$, where $\mathrm{lr}(\alpha) \geq 0$ and is non-increasing along with the index of update cycles α, and satisfy

$$\sum_{\alpha} \mathrm{lr}(\alpha) \to \infty \text{ and } \sum_{\alpha} \mathrm{lr}^2(\alpha) < \infty. \tag{19}$$

Notably, Assumption 1 holds easily. $\sum_{\alpha} \mathrm{lr}(\alpha) \to \infty$ ensures the alternate learning provided with sufficient "time" to thoroughly explore the solution space, thereby theoretically enabling the learning process to converge. $\sum_{\alpha} \mathrm{lr}^2(\alpha) < \infty$ indicates that the squared sums of learning rates are bounded, which helps to reduce oscillations and provide a stable learning process.

Aligned with existing works [44], [45] where two coupled iterations move at different speeds to find a stationary point, we assume that decoder n learns faster than the encoder, to promote the convergence of (5).

Assumption 2: In the alternate learning mechanism, when the encoder updates once, each decoder updates κ times during an update cycle, i.e., $\theta \leftarrow \theta + \mathrm{lr} \cdot \nabla_{\theta} \mathrm{J}(\theta)$, and $\phi_n \leftarrow \phi_n + \sum_{i=1}^{\kappa} \mathrm{lr} \cdot \nabla_{\phi_n^{(i)}} J(\phi_n^{(i)})$, where $\phi_n^{(0)} \leftarrow \phi_n$ and $\phi_n^{(i)} = \phi_n^{(i-1)} + \mathrm{lr} \cdot \nabla_{\phi_n^{(i-1)}} J(\phi_n^{(i-1)})$.

By Assumption 2, we can regard the encoding policy as unchanged and obtain the convergence of the decoder n, as in the following lemma.

Lemma 1: For an independent decoder agent RX_n updating its policy π_{ϕ_n} parameterized by ϕ_n, the state value function converges to a fixed point in $V_{\pi_{\phi_n}}$.

Proof: According to the Bellman equation formulated in [46], one has

$$V_{\pi_{\phi_n}} = R_{\pi_{\phi_n}} + \lambda P_{\pi_{\phi_n}} V_{\pi_{\phi_n}}. \tag{20}$$

We consider a complete metric space $\langle V_{\pi_{\phi_n}}, d \rangle$, as well as a mapping F: $V_{\pi_{\phi_n}} \to V_{\pi_{\phi_n}}$, where d is a metric of nonempty set $V_{\pi_{\phi_n}}$ and satisfies $d : V_{\pi_{\phi_n}} \times V_{\pi_{\phi_n}} \to \mathbb{R}$. For example, choosing an infinity norm metric d, we have $d(V_{\pi_{\phi_n}}^{(i)}, V_{\pi_{\phi_n}}^{(j)}) = \|V_{\pi_{\phi_n}}^{(i)} - V_{\pi_{\phi_n}}^{(j)}\|_{\infty}$, $\forall\, i, j \in \{1, \ldots, \kappa\}$. Meanwhile, for any $\boldsymbol{x}$ in $V_{\pi_{\phi_n}}$, (20) can be re-written as $F(\boldsymbol{x}) = R_{\pi_{\phi_n}} + \lambda P_{\pi_{\phi_n}} \boldsymbol{x}$. Hence, we can observe that

$$\begin{aligned} & d(F(V_{\pi_{\phi_n}}^{(i)}), F(V_{\pi_{\phi_n}}^{(j)})) \\ &= \|(R_{\pi_{\phi_n}} + \lambda P_{\pi_{\phi_n}} V_{\pi_{\phi_n}}^{(i)}) - (R_{\pi_{\phi_n}} + \lambda P_{\pi_{\phi_n}} V_{\pi_{\phi_n}}^{(j)})\|_{\infty} \\ &= \|\lambda P_{\pi_{\phi_n}} (V_{\pi_{\phi_n}}^{(i)} - V_{\pi_{\phi_n}}^{(j)})\|_{\infty} \leq \|\lambda d(V_{\pi_{\phi_n}}^{(i)}, V_{\pi_{\phi_n}}^{(j)})\|_{\infty} \\ &= \lambda d(V_{\pi_{\phi_n}}^{(i)}, V_{\pi_{\phi_n}}^{(j)}). \end{aligned} \tag{21}$$

Then according to the contracting mapping theorem (also known as the Banach Fixed-Point Theorem) [47], we can prove $F(\boldsymbol{x}) = R_{\pi_{\phi_n}} + \lambda P_{\pi_{\phi_n}} \boldsymbol{x}$ has a fixed point in $V_{\pi_{\phi_n}}$, which satisfies $V_{\pi_{\phi_n}} = R_{\pi_{\phi_n}} + \lambda P_{\pi_{\phi_n}} V_{\pi_{\phi_n}}$. As such, we conclude that, iterating continuously from any state, the value function converges to a fixed point in $V_{\pi_{\phi_n}}$. Since our objective function (10) aims to maximize the state value function from $s_n^{(0)}$, which satisfies the above conclusion. □

Based on Lemma 1, we present the following Theorem 3 to address the convergence of the encoder.

Theorem 3: Assuming that in each update cycle, i.e., the encoder updates once while the decoder undergoes κ updates, which are assumed to be sufficient for all decoders converging to a fixed point, the encoder could ultimately reach the convergence as well.

Proof: For the encoder learning, Bellman equation is supervised by Gaussian policy π_{θ}, that is $V_{\pi_{\theta}} = R_{\pi_{\theta}} + \lambda P_{\pi_{\theta}} V_{\pi_{\theta}}$. Without loss of generality, denote α and β as two update cycle indices, while $V_{\pi_{\theta}}^{(\alpha)}$ and $V_{\pi_{\theta}}^{(\beta)}$ ($V_{\pi_{\phi_n}}^{((\alpha-1)\kappa)}$ and $V_{\pi_{\phi_n}}^{((\beta-1)\kappa)}$) are the corresponding learned value functions at the encoder (decoder). If we define $F(\boldsymbol{x}) = R_{\pi_{\theta}} + \lambda P_{\pi_{\theta}} \boldsymbol{x}$, then one has (22), as shown at the bottom of the next page. Specifically, inequality (a) comes from the triangle inequality, and for equality (b), the first term is due to the policy π_{ϕ_n} making the decision, while similarly, the second term stems from the pivotal role played by policy π_{θ}. Thus, according to the contracting mapping theorem [47] and Lemma 1, we get the theorem. □

Remark: As implied by Theorem 3, we optimize the encoder and decoders asynchronously with two different iteration counts, i.e., two coupled iterations that the decoders at RXs update more frequently while the encoder at TX updates at a lower speed. Convergence of these interleaved iterations can be ensured by assuming that the update frequency of the encoder is considerably smaller (i.e., κ smaller) than decoders, which allows decoders to converge first while being perturbed by the slower encoder [44]. Furthermore, the simulation results of Fig. 7(b) (in Section V-B.5) also verify these findings.

V. Performance Evaluation

In this section, we compare the proposed SemanticBC-SCAL with the traditional reliable communication scheme and typical semantic communication algorithms under both AWGN and Rayleigh fading channels respectively. In addition, we also offer an ablation study to comprehensively evaluate and analyze the performance.

A. Simulation Settings

Two datasets are adopted in our experiment: The *standard English version of the proceedings of the European Parliament* [48] consists of $1,136,816$ sentences and $32,478$ words (dictionary size), while the *IMDB dataset* for the sentiment analysis task includes around 0.23 million positive and negative sentence pairs after pre-processing, with a vocabulary that extends beyond $70,000$ words. Subsequently, both datasets are then divided into training and testing with a ratio of $4 : 1$. In our experiment, these two datasets are

TABLE III
NETWORK STRUCTURE AND HYPERPARAMETERS USED IN SEMANITCBC-SCAL

	Modules	Layer	Dimensions	Activation
	Input	$\boldsymbol{m}$	30	\
TX	$SC_{\text{en}}(\cdot)$	Embedding	(30×128)	\
		UT encoder	(30×128)	\
		Dense layer	(30×16)	ReLU
		Gaussian policy	(30×16)	\
	Q	Dense layer	(30×30)	ReLU
		Dense layer	(30×30)	\
Channel	AWGN	\	\	\
3 RXs	$Q_n^{-1}(\cdot)$	Dense layer	(30×16)	ReLU
	$SC_{n,\text{de}}(\cdot)$	Dense layer	(30×128)	ReLU
		UT decoder	(30×128)	ReLU
		Dense layer	$(30 \times 32, 478)$	Dictionary
		Softmax	$(32, 478 \times \hat{T}_n)$	Softmax

pre-processed into sentences with the length of $4 \sim 30$ words, those shorter than 30 words are padded with zeros and the number of units D in the hidden layer behind each word equals 128. Furthermore, the backbone of the semantic encoder/decoders in SemanticBC-SCAL is composed of UT [22] while the channel encoder and decoder are based on dense layers. Specially, UT employs an adaptive computation time (ACT) mechanism to dynamically adjust the number of computation steps required for processing each input token, which leads to more efficient semantic extraction, as simpler tokens can be processed with fewer steps, while more complex tokens involve additional steps [6], [49]. The detailed DNN structure and hyperparameters are summarized in Table III.

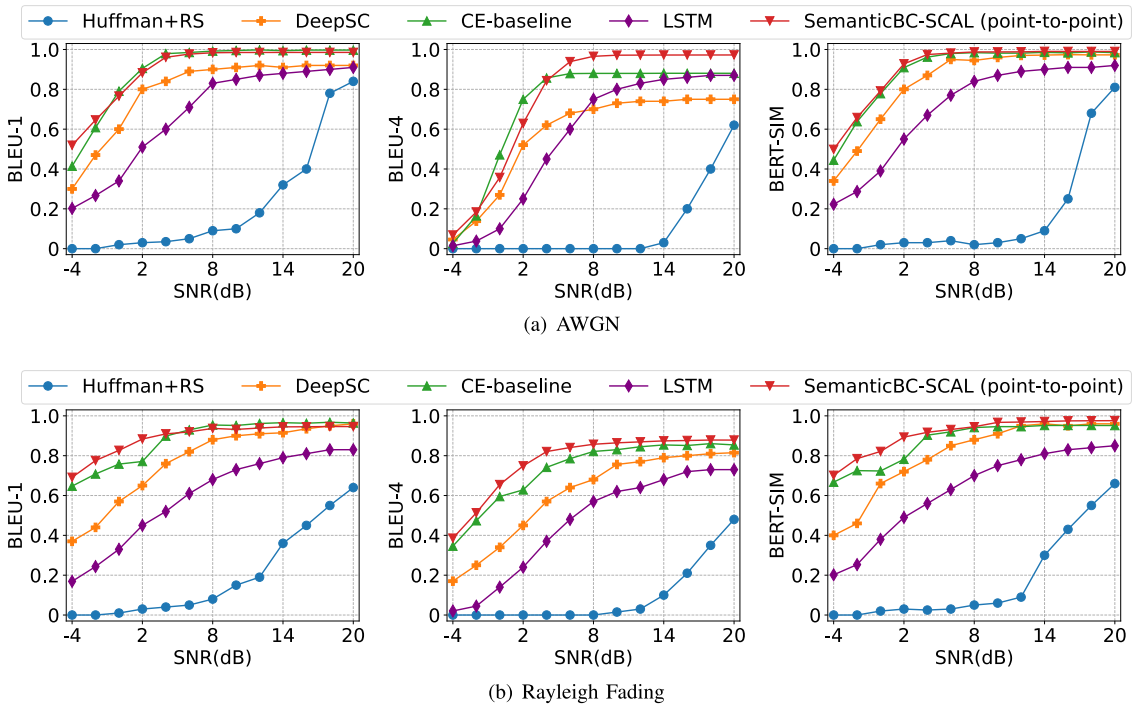
As for the wireless channel, we consider the commonly used AWGN and Rayleigh fading channel as discussed in Section III-A. For comparison, we consider the following baselines to broadcast common or private messages.

1) Huffman-RS: A traditional communication system that source coding and channel coding are implemented as Huffman coding and Reed-Solomon (RS) [25] coding, respectively. Meanwhile, the coding length with RS is set to (30×42) in 64-QAM.
2) CE baseline: It refers to a point-to-point model that shares the same network structure as our proposed model. Put differently, a special case arises when SemanticBC-SCAL is optimized by the CE function.
3) LSTM: It optimizes the entire semantic system on top of RL but its backbone is built around LSTM (Long Short-Term Memory) networks [8].
4) DeepSC: It belongs to pioneering work in semantic communication [5], and leverages the transformer architecture optimized by the CE function and mutual information.
5) MR_DeepSC: It is a one-to-many semantic BC model for multi-user scenario [19], and its backbone is built on DeepSC.
6) DL-JSCC: It develops bidirectional gated recurrent unit (BGRU) based encoder and bidirectional attention mechanism integrated decoders for multi-user BC system [18].
7) SemanticBC-CE: It shares the same structure as our proposed SemanticBC-SCAL, but is optimized by the CE function in the JSCC manner.

Moreover, for SemanticBC-SCAL, the pre-training process is implemented by utilizing the CE function to accelerate the training, since in the context of decision-making for textual sequences, the dimension of the action space exceeding 10^4 poses considerable challenges to converge and makes the direct training computationally prohibitive. Therefore, inspired by the pre-training methods for RL outlined in [43] and [50], we employ supervised learning to steer the model parameters toward reasonable, though not fully converged, initial values. After pre-training for E_p epochs, all pre-trained DNNs undergo the described training procedure in SemanticBC-SCAL until convergence. Besides, to characterize the inherently noisy channel environment, we have implemented two types of SNR simulation methods. Firstly, for the point-to-point case, we set the training SNR to a fixed mean of 10 dB and a variance of 1 dB for each receiver to indicate practical channel conditions. Secondly, in broadcast cases with common messages, we add different noise variances δ^2_{SNR} of SNR for different RXs to reflect their heterogeneous estimation capabilities and adaptability. Specially, for both AWGN and Rayleigh fading channels, we set different mean values μ_{SNR} and noise variance δ^2_{SNR} during the training stage respectively, including $\mu_{\text{SNR},1} = 6$ dB, $\delta_{\text{SNR},1} = 1$ dB; $\mu_{\text{SNR},2} = 10$ dB, $\delta_{\text{SNR},2} = 1$ dB; $\mu_{\text{SNR},3} = 10$ dB, $\delta_{\text{SNR},3} = 2$ dB. While for SemanticBC-SCAL in the point-to-point scenario, we set the SNR of the training process by default with $\mu_{\text{SNR}} = 10$ dB and $\delta_{\text{SNR}} = 1$ dB. The related parameters are listed in Table IV.

To be consistent with existing works [5], [51] on SemCom, we adopt BLEU scores [38], BERT-SIM [39], and WAR to evaluate the system's performance. In particular, BLEU scores count the similarity of n-gram phrases from the recovered

$$
\begin{aligned}
& d(F(V_{\pi_\theta}^{(\alpha)}, V_{\pi_{\phi_n}}^{((\alpha-1)\kappa)}), F(V_{\pi_\theta}^{(\beta)}, V_{\pi_{\phi_n}}^{((\beta-1)\kappa)})) \\
&= \|F(V_{\pi_\theta}^{(\alpha)}, V_{\pi_{\phi_n}}^{((\alpha-1)\kappa)}) - F(V_{\pi_\theta}^{(\alpha)}, V_{\pi_{\phi_n}}^{((\beta-1)\kappa)}) + F(V_{\pi_\theta}^{(\alpha)}, V_{\pi_{\phi_n}}^{((\beta-1)\kappa)}) \\
&\quad - F(V_{\pi_\theta}^{(\beta)}, V_{\pi_{\phi_n}}^{((\beta-1)\kappa)})\|_\infty \overset{(a)}{\leq} \|F(V_{\pi_\theta}^{(\alpha)}, V_{\pi_{\phi_n}}^{((\alpha-1)\kappa)}) - F(V_{\pi_\theta}^{(\alpha)}, V_{\pi_{\phi_n}}^{((\beta-1)\kappa)})\|_\infty + \|F(V_{\pi_\theta}^{(\alpha)}, V_{\pi_{\phi_n}}^{((\beta-1)\kappa)}) \\
&\quad - F(V_{\pi_\theta}^{(\beta)}, V_{\pi_{\phi_n}}^{((\beta-1)\kappa)})\|_\infty \overset{(b)}{=} \|F(V_{\pi_{\phi_n}}^{((\alpha-1)\kappa)}) - F(V_{\pi_{\phi_n}}^{((\beta-1)\kappa)})\|_\infty + \|F(V_{\pi_\theta}^{(\alpha)}) - F(V_{\pi_\theta}^{(\beta)})\|_\infty \\
&= \lambda_1 d(V_{\pi_{\phi_n}}^{((\alpha-1)\kappa)}, V_{\pi_{\phi_n}}^{((\beta-1)\kappa)}) + \|(R_{\pi_\theta} + \lambda_2 P_{\pi_\theta} V_{\pi_\theta}^{(\alpha)}) - (R_{\pi_\theta} + \lambda_2 P_{\pi_\theta} V_{\pi_\theta}^{(\beta)})\|_\infty = \lambda_1 d(V_{\pi_{\phi_n}}^{((\alpha-1)\kappa)}, V_{\pi_{\phi_n}}^{((\beta-1)\kappa)}) \\
&\quad + \lambda_2 \|P_{\pi_\theta}(V_{\pi_\theta}^{(\alpha)} - V_{\pi_\theta}^{(\beta)})\|_\infty \leq \lambda_1 d(V_{\pi_{\phi_n}}^{((\alpha-1)\kappa)}, V_{\pi_{\phi_n}}^{((\beta-1)\kappa)}) + \lambda_2 d(V_{\pi_\theta}^{(\alpha)}, V_{\pi_\theta}^{(\beta)})
\end{aligned} \tag{22}
$$



(a) AWGN

(b) Rayleigh Fading

Fig. 4. BLEU scores and BERT-SIM versus SNR in AWGN and Rayleigh fading channels for the point-to-point case, with the training SNR fixed at a mean of 10 dB and a variance of 1 dB.

TABLE IV
THE DEFAULT PARAMETERS IN SEMANITCBC-SCAL

Parameters	Description	Value
\	Batch size	64
lr	Learning rate	$1e-5$
K	Number of parallel samples	5
κ	Local iterations for decoders in each *update cycle*	$1,000$
E_p	No. of pre-training epochs at TX and RXs	50, 30
E_e	No. of training epochs at TX and RXs	180, 200

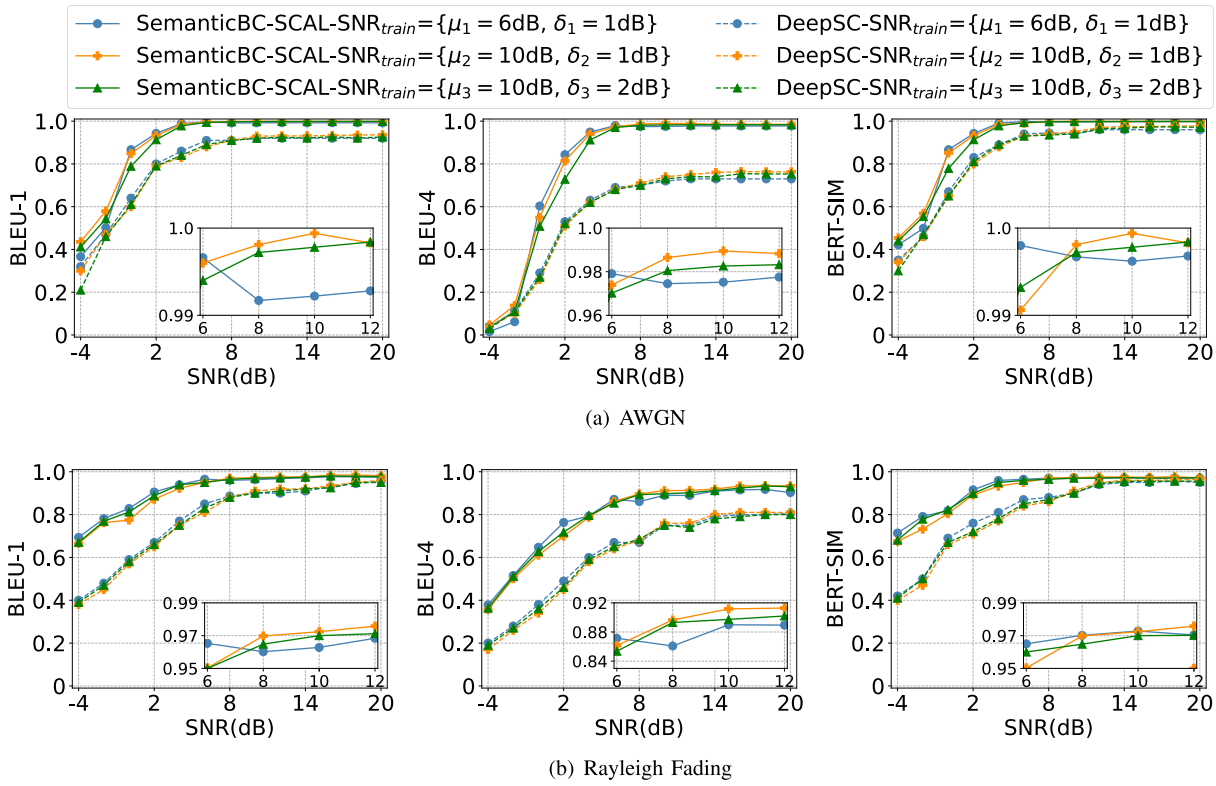
text and the referred ground truth, which can be typically denoted as BLEU-1, BLEU-2, BLEU-3, and BLEU-4 respectively. BERT-SIM calculates the cosine similarity of BERT embeddings after fine-tuning. In some sense, WAR shares some similarity with 1-gram. Nevertheless, BLEU scores [38] consider the length of sentences by introducing a brevity penalty. In other words, if the length of decoded sentences doesn't perfectly match the reference sentence, the computed BLEU score will be penalized. In terms of semantic similarity metric Θ for reward training, we choose the BLEU score by combining 1-gram, 2-gram, 3-gram, and 4-gram settings with equal weights 0.25, so as to provide semantic level supervision.

Besides, as the number of RXs changes, we provide two approaches to address scalability issues (i.e., *Q1*) depending on the availability of computational resources. With sufficient computational resources, the broadcast model can be directly trained with the desired number of RXs using self-critical learning, starting from parameters initially pre-trained with the CE function. The performance achieved through this method can be considered as *upper bound* for a corresponding number of RXs in SemanticBC-SCAL. Alternately, in resource-constrained scenarios, fine-tuning a trained broadcast model becomes preferred, i.e., when the number of RXs decreases, only the TX is required to be fine-tuned to accommodate the change. Conversely, when new RXs are added, the parameters of both the TX and newly added RXs need to be fine-tuned. By default, direct training is applied unless otherwise specified.

B. Numerical Results and Analysis

1) Performance in Point-to-Point SemCom: We start with the performance comparison between the SemanticBC-SCAL with other baselines in the point-to-point case. In particular, we provide the performance in terms of BLEU scores and BERT-SIM under AWGN and Rayleigh fading channel in Fig. 4. As shown in Fig. 4(a), our proposed SemanticBC-SCAL outperforms CE baseline and LSTM, especially in terms of BLEU-4. Since a larger n-gram phrase carries more contextual information, the result demonstrates our approach is capable of preserving semantics and prone to provide a semantic level transmission. Furthermore, our approach also outperforms baselines in terms of BERT-SIM, allowing for accommodating some synonyms. Meanwhile, as illustrated in Fig. 4(b), attributed to the volatile channel gain, the Rayleigh fading channel imposes a stronger detrimental impact compared to the AWGN channel on both the traditional method and SemCom models, leading to a more pronounced attenuation during the signal transmission.

Furthermore, our method exhibits significant superiority over the traditional Huffman-RS method, and stands out among other DNN-based SemCom approaches, especially



(a) AWGN

(b) Rayleigh Fading

Fig. 5. Comparison of SemanticBC-SCAL and DeepSC in terms of BLEU-1 score, BLEU-4 score, and BERT-SIM under both AWGN and Rayleigh fading channels.

in low SNRs. Most impressively, our approach obtains a decoding accuracy that closely approximates 100% at around $\mathrm{SNR} = 8$ dB in AWGN channels. This improved performance can be primarily attributed to the adaptive transformer layer in the UT, which has the ability to reconstruct partial semantic information in a poor channel environment by leveraging shared background knowledge, thereby mitigating the distortion introduced by channel noise. Meanwhile, by adopting semantic similarity scores as rewards, our approach fosters an environment conducive to the system's learning and understanding of semantic representations at a semantic level, which addresses the $Q2$.

Notably, even if our approach has a lower BLEU-1 compared to the CE-baseline in Fig. 4, the existence of synonyms in the recovered sentences actually leads to a slightly higher sentence similarity in terms of BLEU-4. More importantly, although BERT-SIM is not directly adopted as a reward function, the semantic level supervision under BLEU score also leads to the superiority of SemanticBC-SCAL in the point-to-point case and manifests its effectiveness.

2) Performance in Semantic BC With Common Messages: In this case, we assume that all RXs attempt to decode the same common messages. In addition to the inherent channel noise as discussed in (3), we also consider the estimation variance related to SNRs to better simulate the varying conditions. Consequently, during the training process, we employ lower SNR to acclimate to these demanding channel conditions, i.e., exemplified by scenarios, such as the AWGN and Rayleigh channel under $\mu_{\mathrm{SNR},1} = 6$ dB, $\delta_{\mathrm{SNR},1} = 1$ dB. Conversely, in a favorable channel environment, we frequently employ higher SNR settings during training to guarantee robust performance across a substantial portion of the channel environment while maintaining a competitive level of resilience. Fig. 5 visualizes the corresponding BLEU-1, BLEU-4, and BERT-SIM in Semantic BC with three different receivers on both AWGN and Rayleigh channels.

From Fig. 5, it can be observed that our SemanticBC-SCAL demonstrates adaptability to the varying SNR, particularly excelling in low SNR regime. Notably, when the test SNR ranges from 0 to 6 dB, the RX_n trained under low SNR (i.e., $\mu_{\mathrm{SNR},1} = 6$ dB, $\delta_{\mathrm{SNR},1} = 1$ dB) yields superior results than others for both AWGN and Rayleigh fading channels, while DeepSC in BC scenarios follows a similar trend, but has an inferior performance. To elaborate, when the mean value of training SNR is configured at $\mu_{\mathrm{SNR},1} = 10$ dB, for SemanticBC-SCAL in the AWGN channel, the training SNR with a narrower variance (i.e., $\delta_{\mathrm{SNR},1} = 1$ dB) outperforms that with a wider variance (i.e., $\delta_{\mathrm{SNR},1} = 2$ dB) in a low SNR regime, while in Rayleigh fading channels, the opposite is true, thanks to its generalization capabilities. Meanwhile, in the higher SNR regime, the performance of the channel trained under low SNR exhibits insubstantial improvement and the channel with lower variance estimation performs better.

Additionally, we also validate the scalability of our semantic BC system in scenarios with a varying number of RXs, as depicted in Table V. It is noteworthy that as the number of RX increases, there is a modest decrease in semantic decoding proficiency, but overall BLEU scores and WAR consistently maintain a high level. Furthermore, compared to the point-to-point system, our SemanticBC-SCAL demonstrates a notable

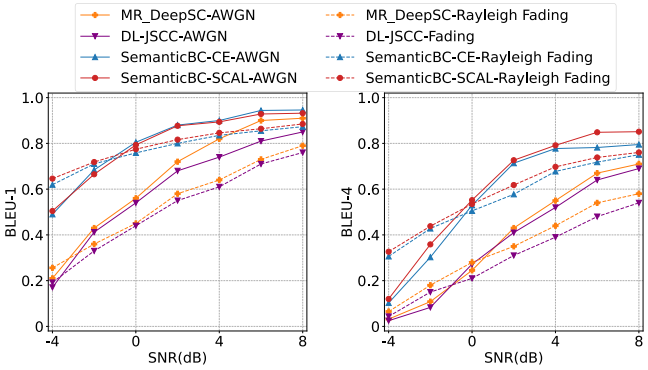


Fig. 6. Comparison of three semantic BC schemes with private messages in terms of BLEU-1 score and BLEU-4 score under both AWGN and Rayleigh fading channels.

TABLE V

COMPARISONS ON THE VARYING NUMBER OF RXS OF SEMANTICBC-SCAL UNDER AWGN CHANNEL, WHEREIN ALL BLEU SCORES AND WAR DENOTE THE AVERAGES WHEN SNR= 10 DB

No. of RXs	BLEU-1	BLEU-2	BLEU-3	BLEU-4	WAR
1	0.9950	0.9904	0.9850	0.9710	0.9876
3	0.9984	0.9969	0.9952	0.9832	0.9964
5	0.9969	0.9947	0.9920	0.9790	0.9839
7	0.9850	0.9804	0.9750	0.9610	0.9776

advantage in utilizing semantics within the broadcast scenario. This advantage can be attributed to the localized adaptive learning and comprehensive integration of all decoders' outcomes. This indicates that our SemanticBC-SCAL can be effectively extended to semantic BC systems with varying numbers of RXs, thereby addressing the $Q1$.

3) Extension to Semantic BC With Private Messages: Considering a broader case of semantic broadcast communication in which a transmitter sends private messages to different receivers, we empirically hypothesize that by slightly adjusting the transmission content during the training phase, it can significantly facilitate the customization of private messages for each receiver. To validate the feasibility of this viewpoint, we provide a practical example in this part and also give a comparison with other semantic BC methods trained by the IMDB dataset.

Specially, at the transmitter side, the interested content may be different for multi-receivers. For example, for a semantic BC with 2 RXs, RX_1 receives positive text whereas RX_2 receives the negative text. In this case, these unique data are individually encoded and broadcast to different RXs. For the receiver side, the RXs reconstruct the intended data according to sentiment features. Notably, when the transmitter broadcasts private messages, it is expected that the encoder can adaptively encode sentiment text to better assist the RXs in enhancing semantic accuracy. Fig. 6 compares the effects of varying SNRs on BLEU-1 and BLEU-4 scores within two RXs under AWGN and Rayleigh fading channels, in which the baselines consist of MR_DeepSC [19], DL-JSCC [18], and SemanticBC-CE. From Fig. 6, SemanticBC-SCAL always outperforms other models in terms of the BLEU-4 score, which implies that our proposed method can effectively understand and represent semantic information.

TABLE VI

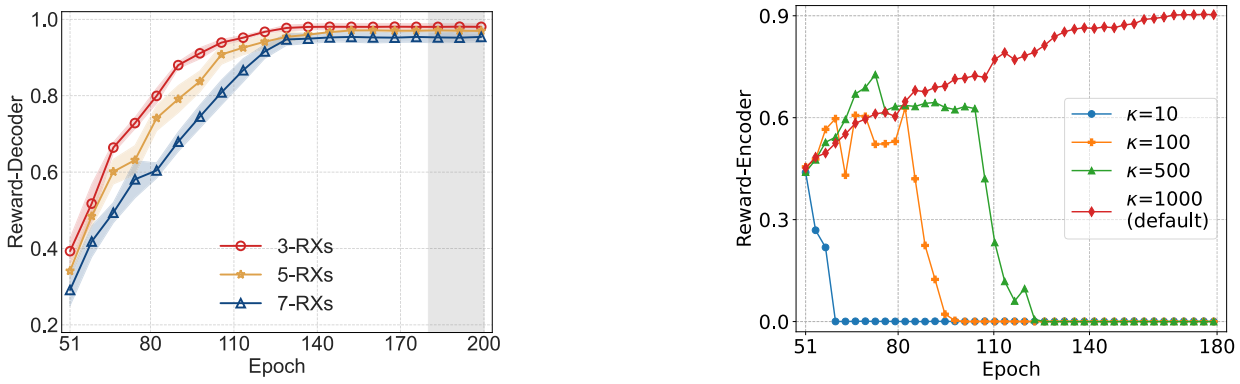
COMPUTATIONAL COMPLEXITY FOR DIFFERENT SEMANTIC BC METHODS

Methods	Complexity	
	Additions	Multiplications
SemanticBC-SCAL	5.4×10^5	2.7×10^7
DeepSC	5.2×10^5	2.1×10^7
MR_DeepSC	6.3×10^7	6.3×10^7
DL-JSCC	5.1×10^5	1.3×10^8

Additionally, we observe that the performance of all models varies across different datasets. In particular, the outcomes on the IMDB dataset are marginally lower than those on the European Parliament dataset. This discrepancy likely arises from the significantly smaller size of the IMDB dataset, i.e., the processed IMDB dataset consists of 0.23 million sentence pairs, a quantity substantially less than the 2 million sentences in the European Parliament dataset, indicating that limited training data can restrict the upper bound of decoders' performance.

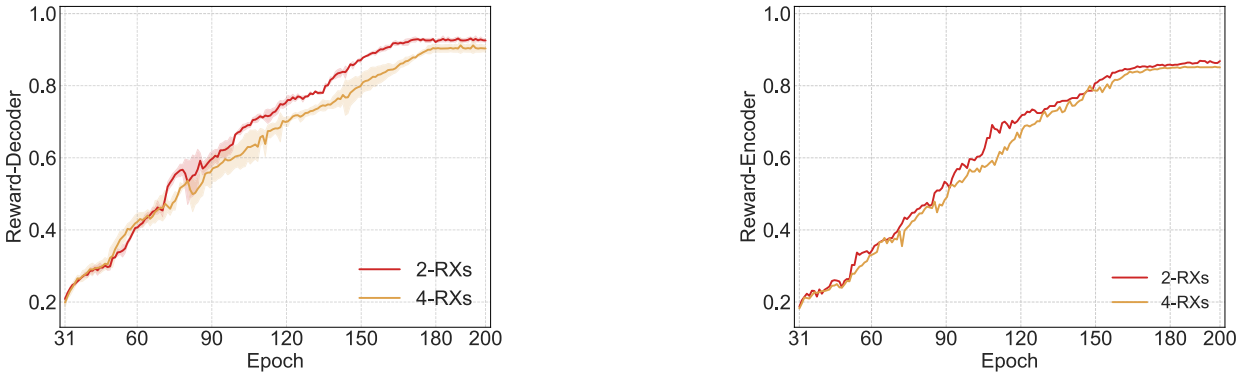
4) Computational Complexity: The computational complexity for different methods is compared in Table VI. Our primary analysis focuses on the complexities associated with the encoder/decoder, channel layers, and linear layers, revealing that a substantial amount of computational effort is centered on the encoder and decoder. Compared with the DeepSC framework, our proposed SemanticBC-SCAL exhibits marginally increased computational complexity. This increment is attributable to the utilization of the UT architecture within our encoder-decoder configuration, which involves a dynamic adjustment in the number of transformer layers. The empirical analysis indicates the average transformer layer that our model requires is 3.5 layers, in contrast to the static three-layer architecture employed by DeepSC. In MR_DeepSC, each receiver employs a DistilBERT-based recognizer to distinguish users, which consequently leads to a slight increase in computational complexity. DL-JSCC requires more multiplications than other DNN-based models due to the complicated structure of BGRU and GRU. Overall, despite a slight increase in computational complexity, our proposed method yields a significant improvement in performance, thus achieving a better balance between complexity and performance.

5) Convergence of SemanticBC-SCAL: To testify whether our alternate learning mechanism guarantees convergence, we have depicted the evolution of the reward for the encoder and decoders over epochs in Fig. 7. From Fig. 7(a), we can observe that consistent with Lemma 1, our alternative learning mechanism eventually converges with steady learning and low variance (among different receivers) for the different numbers of RXs, which promises an effective learning approach for the large-scale knowledge base. Besides, we also validate whether the performance can be further improved if each decoder updates its parameters independently, and the shaded region in Fig. 7(a) represents the results of an additional 20 epochs



(a) (b)

Fig. 7. Convergence of SemanticBC-SCAL with common messages, wherein the reward curves are the average BLEU score for all broadcast receivers. (a) The training reward of the decoder sides with varying numbers of RXs, where the shaded region represents an additional 20 training epochs to update the decoders' parameters independently. (b) The training reward of the encoder side with varying local iterations κ under three RXs.



(a) (b)

Fig. 8. Convergence of SemanticBC-SCAL with private messages, wherein the reward curves are the average BLEU score for all broadcast RXs. (a) The training reward of the decoder sides with varying numbers of RXs. (b) The training reward of the encoder side with varying numbers of RXs.

of local training on the decoders. This suggests that further refinements to the decoders independently after alternating training do not significantly alter the performance, thereby corroborating the robustness and reliability of our approach.

Furthermore, as highlighted in the "Remark" accompanying Theorem 3, the convergence of these interleaved iterations between one encoder and multiple decoders can be ensured by certain locally updated iterations κ. Consequently, we delve into exploring the influence of κ on the convergence of the encoder, as illustrated in Fig. 7(b). Based on the observations from Fig. 7(b), it becomes apparent that when κ is relatively small, indicating a low number of local iterations for decoders and frequent updates for the encoder per epoch. In this sense, the encoder fails to learn effectively even after a few training epochs. As κ increases, the encoder can sustain longer training epochs but still struggles to converge, until $\kappa = 1000$. Only at this point, it achieves continuous and stable learning, eventually achieving convergence, which aligns with those presented in Theorem 3. Furthermore, we also validate the convergence of our method that involves private messages. As shown in Fig. 8, as the number of RXs increases, there is a modest decrease in semantic decoding proficiency, but the overall performance consistently maintains a high level and tends to be convergent. In this way, these empirical findings address

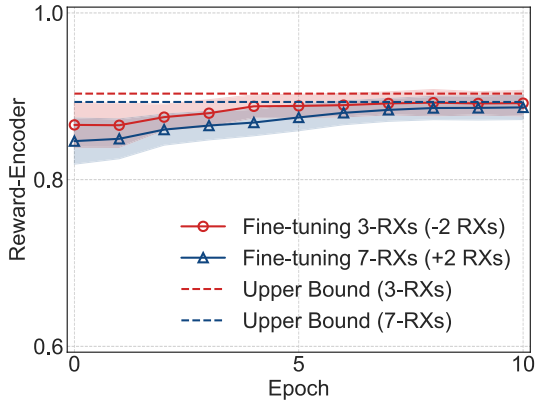


Fig. 9. The convergence of SemanticBC-SCAL fine-tuning with an increase ($+2$ RXs) or decrease (-2 RXs) in the number of RXs for common messages transmission, wherein the initially pre-trained semantic broadcast model has 5 RXs.

$Q3$. Moreover, in resource-constrained scenarios, fine-tuning a pre-trained model becomes preferred. For example, as shown in Fig. 9, starting from the trained SemanticBC-SCAL model with 5 RXs, it only needs about 7 epochs' fine-tuning of the encoder only to accommodate a 3-RX case and nearly approach the upper-bound performance. Similarly, for an

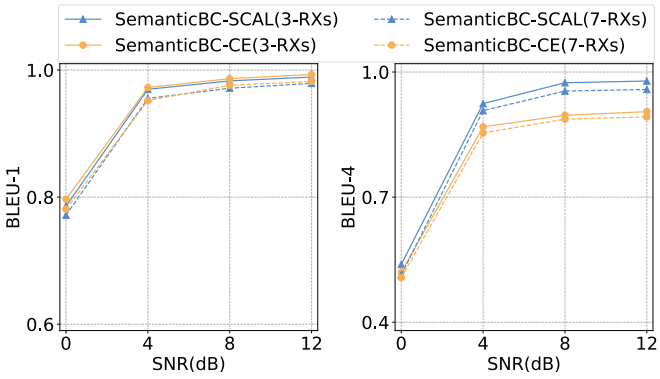


Fig. 10. The fine-tuning performance comparison of SemanticBC-SCAL and SemanticBC-CE with common messages in terms of BLEU-1 score and BLEU-4 score under the AWGN channel.

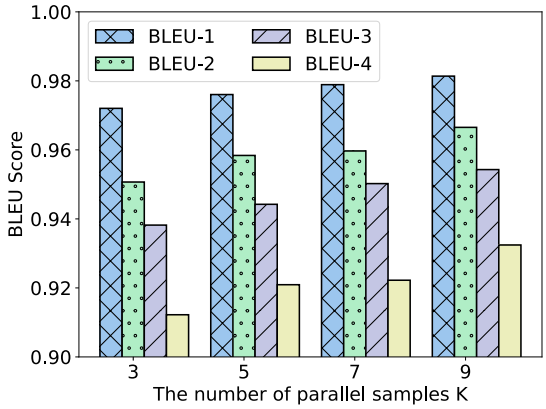


Fig. 11. The impact of parallel samples K with common messages in the SemanticBC-SCAL system, wherein all RXs share the common messages.

increase in the number of RXs by 2, it requires only about 8 epochs to fine-tune the encoder and the newly added RXs to achieve performance nearly up to that of direct training. Furthermore, in Fig. 10, we also compare the semantic similarity performance achieved after fine-tuning SemanticBC-SCAL and SemanticBC-CE. It can be observed that, although SemanticBC-CE behaves almost similarly at BLEU-1 score, our alternating learning approach performs better in terms of the BLEU-4 score, which reflects semantic similarity at the semantic level more accurately. In a nutshell, these empirical results validate the effectiveness of the alternating learning approach, which competently addresses the scalability and convergence issues, i.e., $Q1$ and $Q3$.

6) Ablation Study: The ablation study for comparing different numbers of parallel samples K in (12) and (16) is given in Fig. 11. According to (12) and (16), as K increases, the mean reward from K rollouts of state-action value can be estimated more precisely. Consistently, as depicted in Fig. 11, a larger K obtains a higher semantic similarity, which is consistent with [40].

VI. Conclusion

In this paper, to cope with scalability, compatibility, and convergence issues in semantic broadcast communications, we have proposed a semantic broadcast communication framework optimized by an RL-based self-critical alternate learning, called SemanticBC-SCAL, which provides semantic level supervision and improves the robustness under varying numbers of RXs. In particular, we have regarded semantic similarity as a reward and formulated the optimization process as an RL problem. Furthermore, we have adopted self-critical learning to obtain a simple and efficient gradient estimation, especially suitable for complex SemCom systems with large-scale samples. To accommodate varying numbers of RXs and address non-differentiable channels, we have adopted a cost-effective alternate training mechanism to asynchronously learn the encoder and multiple decoders with different learning iterations instead of directly backpropagating gradients through the channel. Moreover, we have also delved into the convergence analysis of SemanticBC-SCAL and provided conditions for the convergence. Extensive simulation results with both common and private messages have demonstrated that our proposed semantic broadcast communication system can achieve high semantic accuracy with different numbers of RXs, and has exhibited strong scalability and robustness for different channel environments. Moreover, since sharing a consistent structure among different RXs limits some applications, for future works, we aim to further explore the possibility of developing scalable receiver structures, and enhance the diversity of tasks in more general scenarios.

Acknowledgment

The authors would like to thank Kun Lu, a class-2023 graduate from the NICE Laboratory, Zhejiang University, for his foundational codes[2] and helpful discussions.

References

[1] C.-X. Wang et al., "On the road to 6G: Visions, requirements, key technologies and testbeds," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 2, pp. 905–974, 2nd Quart., 2023.

[2] W. Weaver, "Recent contributions to the mathematical theory of communication," *Math. Theory Commun.*, vol. 10, no. 4, pp. 261–281, Sep. 1949.

[3] Z. Lu et al., "Semantics-empowered communications: A tutorial-cum-survey," *IEEE Commun. Surveys Tuts.*, vol. 26, no. 1, pp. 41–79, 1st Quart., 2024.

[4] N. Farsad, M. Rao, and A. Goldsmith, "Deep learning for joint source-channel coding of text," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Calgary, AB, Canada, Apr. 2018, pp. 2326–2330.

[5] 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, 2021.

[6] Q. Zhou, R. Li, Z. Zhao, C. Peng, and H. Zhang, "Semantic communication with adaptive universal transformer," *IEEE Wireless Commun. Lett.*, vol. 11, no. 3, pp. 453–457, Mar. 2021.

[7] M. Sana and E. C. Strinati, "Learning semantics: An opportunity for effective 6G communications," in *Proc. IEEE 19th Annu. Consum. Commun. Netw. Conf. (CCNC)*, Las Vegas, NV, USA, Jan. 2022, pp. 631–636.

[8] K. Lu, R. Li, X. Chen, Z. Zhao, and H. Zhang, "Reinforcement learning-powered semantic communication via semantic similarity," 2021, *arXiv:2108.12121*.

[9] Z. Weng and Z. Qin, "Semantic communication systems for speech transmission," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2434–2444, Aug. 2021.

[10] H. Tong et al., "Federated learning for audio semantic communication," *Frontiers Commun. Netw.*, vol. 2, Sep. 2021, Art. no. 734402.

[2] https://github.com/lukun199/semanticrl

[11] 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.

[12] 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, Sep. 2019.

[13] D. B. Kurka and D. Gündüz, "Deep joint source-channel coding of images with feedback," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Barcelona, Spain, May 2020, pp. 5235–5239.

[14] Z. Zhang, Q. Yang, S. He, M. Sun, and J. Chen, "Wireless transmission of images with the assistance of multi-level semantic information," in *Proc. Int. Symp. Wireless Commun. Syst. (ISWCS)*, Hangzhou, China, Oct. 2022, pp. 1–6.

[15] T.-Y. Tung and D. Gunduz, "Deep-learning-aided wireless video transmission," in *Proc. IEEE 23rd Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Oulu, Finland, Jul. 2022, pp. 1–5.

[16] P. Jiang, C.-K. Wen, S. Jin, and G. Y. Li, "Wireless semantic communications for video conferencing," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 1, pp. 230–244, Jan. 2022.

[17] M. Ding, J. Li, M. Ma, and X. Fan, "SNR-adaptive deep joint source-channel coding for wireless image transmission," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Toronto, ON, Canada, Jun. 2021, pp. 1555–1559.

[18] T. Liu and X. Chen, "Attention-based neural joint source-channel coding of text for point to point and broadcast channel," *Artif. Intell. Rev.*, vol. 55, no. 3, pp. 2379–2407, Mar. 2022.

[19] H. Hu, X. Zhu, F. Zhou, W. Wu, R. Q. Hu, and H. Zhu, "One-to-many semantic communication systems: Design, implementation, performance evaluation," *IEEE Commun. Lett.*, vol. 26, no. 12, pp. 2959–2963, Dec. 2022.

[20] S. Ma et al., "Features disentangled semantic broadcast communication networks," 2023, *arXiv:2303.01892*.

[21] V. K. Shrivastava, S. Baek, and Y. Baek, "5G evolution for multicast and broadcast services in 3GPP release 17," *IEEE Commun. Standards Mag.*, vol. 6, no. 3, pp. 70–76, Sep. 2022.

[22] M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit, and Ł. Kaiser, "Universal transformers," 2018, *arXiv:1807.03819*.

[23] R. Luo, "A better variant of self-critical sequence training," 2020, *arXiv:2003.09971*.

[24] H. Xie, Z. Qin, and G. Y. Li, "Semantic communication with memory," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 8, pp. 2658–2669, Aug. 2023.

[25] I. S. Reed and G. Solomon, "Polynomial codes over certain finite fields," *J. Soc. Ind. Appl. Math.*, vol. 8, no. 2, pp. 300–304, Jun. 1960.

[26] H. Xie and Z. Qin, "A lite distributed semantic communication system for Internet of Things," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 142–153, Jan. 2020.

[27] Y. Wang et al., "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.

[28] H. Seo, J. Park, M. Bennis, and M. Debbah, "Semantics-native communication with contextual reasoning," *IEEE Trans. Cogn. Commun. Netw.*, vol. 9, no. 3, pp. 604–617, Jun. 2023.

[29] M. Jankowski, D. Gündüz, and K. Mikolajczyk, "Deep joint source-channel coding for wireless image retrieval," in *Proc. IEEE Int. Conf. Acoust. Speech. Signal. Process. (ICASSP)*, Barcelona, Spain, May 2020, pp. 5070–5074.

[30] M. Jankowski, D. Gündüz, and K. Mikolajczyk, "Wireless image retrieval at the edge," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 89–100, Jan. 2020.

[31] M. Kountouris and N. Pappas, "Semantics-empowered communication for networked intelligent systems," *IEEE Commun. Mag.*, vol. 59, no. 6, pp. 96–102, Jun. 2021.

[32] H. Xie, Z. Qin, X. Tao, and K. B. Letaief, "Task-oriented multi-user semantic communications," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 9, pp. 2584–2597, Sep. 2022.

[33] G. Zhang, Q. Hu, Z. Qin, Y. Cai, G. Yu, and X. Tao, "A unified multi-task semantic communication system for multimodal data," 2022, *arXiv:2209.07689*.

[34] Q. Yuan, X. Fu, Z. Li, G. Luo, J. Li, and F. Yang, "GraphComm: Efficient graph convolutional communication for multiagent cooperation," *IEEE Internet Things J.*, vol. 8, no. 22, pp. 16359–16369, Nov. 2021.

[35] W. J. Yun et al., "Attention-based reinforcement learning for real-time UAV semantic communication," in *Proc. Int. Symp. Wireless Commun. Syst. (ISWCS)*, Berlin, Germany, 2021, pp. 1–6.

[36] T.-Y. Tung, S. Kobus, J. P. Roig, and D. Gündüz, "Effective communications: A joint learning and communication framework for multi-agent reinforcement learning over noisy channels," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2590–2603, Aug. 2021.

[37] X. Li, J. Shi, and Z. Chen, "Task-driven semantic coding via reinforcement learning," *IEEE Trans. Image Process.*, vol. 30, pp. 6307–6320, 2021.

[38] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, Philadelphia, PA, USA, 2002, pp. 311–318.

[39] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol. (NAACL HLT)*, vol. 1, Minneapolis, MI, USA, Jun. 2019, pp. 4171–4186.

[40] J. Gao, S. Wang, S. Wang, S. Ma, and W. Gao, "Self-critical n-step training for image captioning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 6300–6308.

[41] K. A. Dubey, S. J. Reddi, S. A. Williamson, B. Poczos, A. J. Smola, and E. P. Xing, "Variance reduction in stochastic gradient Langevin dynamics," in *Proc. Adv. Neural Inf. Proces. Syst. (NIPS)*, Barcelona, Spain, Dec. 2016, pp. 1162–1170.

[42] O. Dogru, K. Velswamy, and B. Huang, "Actor-critic reinforcement learning and application in developing computer-vision-based interface tracking," *Engineering*, vol. 7, no. 9, pp. 1248–1261, Sep. 2021.

[43] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 7008–7024.

[44] M. Holzleitner, L. Gruber, J. Arjona-Medina, J. Brandstetter, and S. Hochreiter, *Convergence Proof for Actor-Critic Methods Applied to PPO and RUDDER*. Berlin, Heidelberg: Springer, 2021, pp. 105–130.

[45] P. Karmakar and S. Bhatnagar, "Two time-scale stochastic approximation with controlled Markov noise and off-policy temporal-difference learning," *Math. Oper. Res.*, vol. 43, no. 1, pp. 130–151, Feb. 2018.

[46] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[47] R. P. Agarwal, M. Meehan, and D. O'Regan, *Fixed Point Theory and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2001.

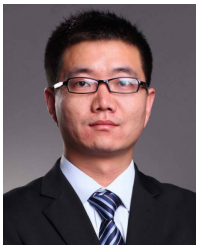
[48] P. Koehn, "Europarl: A parallel corpus for statistical machine translation," in *Proc. AAMT 10th Mach. Transl. Summit.*, Phuket, Thailand, Sep. 2005, pp. 79–86.

[49] B. Wang, R. Li, J. Zhu, Z. Zhao, and H. Zhang, "Knowledge enhanced semantic communication receiver," *IEEE Commun. Lett.*, vol. 27, no. 7, pp. 1794–1798, Jul. 2023.

[50] S. Yun, J. Choi, Y. Yoo, K. Yun, and J. Y. Choi, "Action-decision networks for visual tracking with deep reinforcement learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1349–1358.

[51] P. Jiang, C.-K. Wen, S. Jin, and G. Y. Li, "Deep source-channel coding for sentence semantic transmission with HARQ," *IEEE Trans. Commun.*, vol. 70, no. 8, pp. 5225–5240, Aug. 2022.

Zhilin Lu received the M.S. degree from North China Electric Power University, Beijing, China, in 2021. She is currently pursuing the Ph.D. degree with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China. Her research interests include semantic communication and deep reinforcement learning.



Rongpeng Li (Senior Member, IEEE) was a Research Engineer with the Wireless Communication Laboratory, Huawei Technologies Company Ltd., Shanghai, China, from August 2015 to September 2016. He was also a Visiting Scholar with the Department of Computer Science and Technology, University of Cambridge, Cambridge, U.K., from February 2020 to August 2020. He is currently an Associate Professor with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China. His research interests include networked intelligence for communications evolving (NICE). He received the Wu Wenjun Artificial Intelligence Excellent Youth Award in 2021. He serves as an Editor for *China Communications*.

Ming Lei (Member, IEEE) received the B.Eng. and Ph.D. degrees in information and communication engineering from Zhejiang University, China, in 2006 and 2012, respectively. He is currently an Associate Professor with the College of Information Science and Electronic Engineering, Zhejiang University. His research interests include anti-jamming ad hoc networks, routing strategy, channel estimation and equalization, cooperative communication, and machine learning for wireless communication.

Chan Wang received the B.Eng. degree in information and communication engineering and the Ph.D. degree in microelectronics and solid-state electronics from Zhejiang University, China, in 2012 and 2017, respectively. Since October 2017, she has been with Zhejiang University where she is currently a Post-Doctoral Researcher with the College of Information Science and Electronic Engineering. Her research interests include channel coding, cooperative communication, machine learning for wireless communications, and ad hoc networks.

Zhifeng Zhao (Member, IEEE) received the B.E. degree in computer science, the M.E. degree in communication and information systems, and the Ph.D. degree in communication and information systems from the PLA University of Science and Technology, Nanjing, China, in 1996, 1999, and 2002, respectively. From 2002 to 2004, he was a Post-Doctoral Researcher with Zhejiang University, Hangzhou, China, where his researches were focused on multimedia next-generation networks (NGNs) and softswitch technology for energy efficiency. From 2005 to 2006, he was a Senior Researcher with the PLA University of Science and Technology, where he performed research and development on advanced energy-efficient wireless routers, ad-hoc network simulators, and cognitive mesh networking test-bed. From 2006 to 2019, he was an Associate Professor with the College of Information Science and Electronic Engineering, Zhejiang University. Currently, he is with Zhejiang Laboratory, as the Chief Engineering Officer. His research interests include software-defined networks (SDNs), wireless networks in 6G, computing networks, and collective intelligence. He is the Symposium Co-Chair of ChinaCom 2009 and 2010. He is the Technical Program Committee (TPC) Co-Chair of the 10th IEEE International Symposium on Communication and Information Technology (ISCIT 2010).

Honggang Zhang (Fellow, IEEE) was a Professor with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou, China. He was also an Honorary Visiting Professor with the University of York, York, U.K., and an International Chair Professor of Excellence with the Université Européenne de Bretagne and Supélec, France. He is currently a Professor with the Faculty of Data Science, City University of Macau, Macau, China. He has co-authored and edited two books: *Cognitive Communications: Distributed Artificial Intelligence (DAI), Regulatory Policy and Economics, Implementation* (John Wiley & Sons) and *Green Communications: Theoretical Fundamentals, Algorithms and Applications* (CRC Press). His research interests include cognitive radio networks, semantic communications, green communications, machine learning, artificial intelligence, intelligent computing, and internet of intelligence. He was a co-recipient of the 2021 IEEE Communications Society Outstanding Paper Award and the 2021 IEEE INTERNET OF THINGS JOURNAL Best Paper Award. He was the leading Guest Editor for the Special Issues on Green Communications of *IEEE Communications Magazine*. He served as a Series Editor for *IEEE Communications Magazine* (Green Communications and Computing Networks Series) from 2015 to 2018 and the Chair of the Technical Committee on Cognitive Networks of the IEEE Communications Society from 2011 to 2012. He was the founding Chief Managing Editor of *Intelligent Computing*, a *Science Partner Journal*. He is the Associate Editor-in-Chief of *China Communications*.