CSI-GPT: Integrating Generative Pre-Trained Transformer With Federated-Tuning to Acquire Downlink Massive MIMO Channels

Ye Zeng 🆔, Li Qiao 🆔, *Graduate Student Member, IEEE*,
Zhen Gao 🆔, *Member, IEEE*, Tong Qin, Zhonghuai Wu,
Emad Khalaf 🆔, Sheng Chen 🆔, *Life Fellow, IEEE*,
and Mohsen Guizani 🆔, *Fellow, IEEE*

Abstract—In massive multiple-input multiple-output (MIMO) systems, how to reliably acquire downlink channel state information (CSI) with low overhead is challenging. In this work, by integrating the generative pre-trained Transformer (GPT) with federated-tuning, we propose a CSI-GPT approach to realize efficient downlink CSI acquisition. Specifically, we first propose a Swin Transformer-based channel acquisition network (SWTCAN) to acquire downlink CSI, where pilot signals, downlink channel estimation, and uplink CSI feedback are jointly designed. Furthermore, to solve the problem of insufficient training data, we propose a variational auto-encoder-based channel sample generator (VAE-CSG), which can generate sufficient CSI samples based on a limited number of high-quality CSI data obtained from the current cell. The CSI dataset generated from VAE-CSG will be used for pre-training SWTCAN. To fine-tune the pre-trained SWTCAN for improved performance, we propose an online federated-tuning method, where only a small amount of SWTCAN parameters are unfrozen and updated using over-the-air computation, avoiding the high communication overhead caused by aggregating the complete CSI samples from user equipment (UEs) to the BS for centralized fine-tuning. Simulation results verify the advantages of the proposed SWTCAN and the communication efficiency of the proposed federated-tuning method.

Index Terms—Channel estimation, CSI feedback, federated learning, generative AI, massive MIMO, swin transformer.

Ye Zeng, Li Qiao, and Tong Qin are with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China (e-mail: qiaoli@bit.edu.cn).

Zhen Gao is with the State Key Laboratory of CNS/ATM, Beijing Institute of Technology, Beijing 100081, China, also with the Beijing Institute of Technology, Zhuhai 519088, China, also with the MIT Key Laboratory of Complex-Field IntelligentSensing, Beijing Institute of Technology, Beijing 100081, China, also with the Advanced Technology Research Institute, Beijing Institute of Technology Jinan, Jinan 250307, China, and also with the Yangtze Delta Region Academy, Beijing Institute of Technology Jiaxing, Jiaxing 314019, China (e-mail: gaozhenl6@bit.edu.cn).

Zhonghuai Wu is with the State Key Laboratory of Explosion Science and Safety Protection, Beijing Key Laboratory for Precision Optoelectronic Measurement Instrument and Technology, School of Optics and Photonics, Beijing Institute of Technology, Beijing 100081, China.

Emad Khalaf is with the Electrical and Computer Engineering Department, Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia (e-mail: ekhalaf@kau.edu.sa).

Sheng Chen is with the School of Electronics and Computer Science, University of Southampton, SO17 1BJ Southampton, U.K., and with the King Abdulaziz University, Jeddah 21589, Saudi Arabia (e-mail: sqc@ecs.soton.ac.uk).

Mohsen Guizani is with the Department of Machine Learning, Mohamed bin Zayed University of Artificial Intelligence, Masdar CJJ9+XC8, UAE (e-mail: mguizani@ieee.org).

Our code is publicly available at https://gaozhen16.github.io/
Digital Object Identifier 10.1109/TVT.2024.3493463

I. INTRODUCTION

In massive multiple-input multiple-output (MIMO) systems, accurate downlink channel state information (CSI) is crucial for beamforming and resource allocation. However, in frequency division duplexing (FDD) systems, accurate estimation and feedback of downlink CSI with low pilot/feedback overhead is challenging, due to the high-dimensional CSI caused by the large number of antennas at the base station (BS) and the non-reciprocity between uplink and downlink channels [1], [2], [3], [4].

As the channel gains associated with different antennas are correlated, the massive MIMO CSI matrix has the inherent redundancy, which can be exploited to reduce the CSI acquisition overhead [5], [6], [7], [8]. Due to its powerful capabilities of feature perception and extraction, deep learning (DL) has been widely used to process CSI in massive MIMO systems for various tasks. The authors of [9] proposed a DL-based joint pilot design and channel estimation scheme, where the fully connected (FC) layer and the convolutional neural network (CNN) are used to design the pilot signal and estimate the CSI, respectively. The authors of [10] made improvements to [9] by designing pilot signals on different subcarriers differently to further reduce the pilot overhead. As for CSI feedback, the seminal work [11] proposed an autoencoder (AE)-based end-to-end (E2E) optimization framework, and the authors of [12] improved CSI feedback performance by introducing receptive fields of different sizes for CSI feature extraction. To improve the deployability of CSI feedback, the authors in [13] proposed a scheme in which the length of the feedback codeword is variable with the sparsity of the channels. The authors in [14] proposed a scheme based on knowledge distillation. Both schemes largely reduce the complexity of the network that needs to be deployed. In addition, the latest information about the application of DL in CSI feedback and the comparison can be obtained from [15].

More recently, as reported in [16], [17], joint design of pilot signals, downlink CE and CSI feedback using DL can further improve the downlink CSI acquisition performance. Another approach to improve the performance is to exploit more advanced neural network (NN) architectures. The authors of [18] investigated the application of Transfomer architecture in various massive MIMO processing tasks, which consistently shows advantages over CNN-based algorithms. With the development of NN architectures, variants of Transformer, e.g., Swin Transformer, have shown better feature capture capability in image processing tasks [19], which is another blessing of DL-based massive MIMO signal processing.

The current DL models [9], [10], [11], [12], [16], [17], [18] need a large amount of high-quality CSI samples for training, which can be obtained from actual measurements or generated from classical channel models, e.g., COST 2100 and clustered delay line (CDL) channel [20]. However, it is either difficult or communication-inefficient to obtain a large number of actual CSI samples in various practical scenarios [21], and it may degrade the NN performance if the features of training dataset and test dataset are not consistent. To overcome this issue, the generative adversarial network (GAN) is employed in [21] to generate CSI training datasets based on only a small amount of CSI measurements. Another promising solution [22] is to use federated learning (FL) to avoid collecting high-dimensional actual CSI samples, thereby reducing the communication overhead considerably. By exploiting gradient compression and over-the-air computation (AirComp), the authors of [23] proposed a communication-efficient FL framework for image classification tasks. The authors further proposed a massive
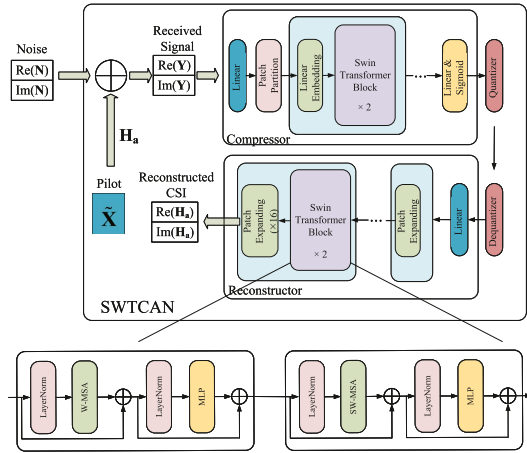
Fig. 1.   Structure of the proposed SWTCAN.

digital AirComp scheme that are compatible with the current wireless networks in [24]. To tune a large NN model, e.g., Transformer, in a communication-efficient manner, the authors of [25] utilized FL to fine-tune part of the parameters of a pre-trained Transformer model. The authors of [26], [27], [28] showed the potential of FL-based CE and/or CSI feedback for massive MIMO systems. They also showed that users' data privacy can be protected by using FL. However, whether FL is more communication-efficient than conventional centralized learning (CL) that requires the feedback of CSI samples from user equipment (UEs) to BS remains unexplored.

In this paper, by integrating the generative pre-trained Transformer (GPT) with federated-tuning, we propose a CSI-GPT approach to realize efficient downlink CSI acquisition. Our main contributions can be summarized as follows.

- We propose a Swin Transformer-based channel acquisition network (SWTCAN) as shown in Fig. 1 to acquire downlink CSI with lower pilot/feedback overhead, where downlink pilot signal, CE and CSI feedback are jointly designed. Our SWTCAN not only retains the extraction capability of the conventional Transformer-based approach [18] but also overcomes its weakness in multi-scale feature extraction. Consequently, lower pilot and feedback overhead can be achieved.
- We propose a variational AE-based channel sample generator (VAE-CSG), which can effectively solve the problem of insufficient high-quality CSI samples. Since channel features in different cells vary dramatically, to maximize the potential of SWTCAN, its training at different BSs should rely on large numbers of CSI samples of the respective cells. However, the number of high-quality CSI samples from the current cell is usually limited. We propose a pre-trained strategy by pre-training VAE-CSG using a large number of CSI samples that typically have different features from those of the current cell. Subsequently, we fine-tune VAE-CSG using a limited number of high-quality CSI samples from the current cell. The fine-tuned VAE-CSG then generates a large number of CSI samples for pre-training SWTCAN.
- Finally, to fine-tune the pre-trained SWTCAN for improved performance, we propose an online federated-tuning method. Only a small amount of SWTCAN parameters (around 11%) are unfrozen and updated using AirComp, avoiding the high communication overhead caused by aggregating the CSI samples from UEs to the BS for centralized fine-tuning. The simulation results demonstrate that in typical cases, the proposed federated-tuning

method can reduce uplink communication overhead by up to 34.8% compared to the traditional CL method.

Notation: Boldface lower and upper-case symbols denote column vectors and matrices, respectively. Superscripts $(\cdot)^{\mathrm{T}}$ and $(\cdot)^{\mathrm{H}}$ denote the transpose and conjugate transpose operators, respectively. $p(\mathbf{x} \mid \mathbf{y})$ is the conditional distribution of $\mathbf{x}$ given $\mathbf{y}$. $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ means $\mathbf{x}$ following a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. $\|\mathbf{A}\|_F$ and $[\mathbf{A}]_{m,n}$ denote the Frobenius norm and the m-th row and n-th column element of the matrix $\mathbf{A}$, respectively. $\|\mathbf{x}\|_p$, $[\mathbf{x}]_m$, and $|\mathbf{x}|_c$ denote the l_p norm, the m-th element, and the cardinality of the vector $\mathbf{x}$, respectively. $\mathbf{I}$ is the identity matrix. $\mathbf{0}$ is a vector with all the elements being 0.

II. PROPOSED SWIN TRANSFORMER-BASED DOWNLINK CSI ACQUISITION SCHEME

A. System Model

We assume that the BS deploys a uniform planar array (UPA) with N_{BS} antennas to serve U single-antenna UEs in an FDD mode. Orthogonal frequency division multiplexing (OFDM) with P subcarriers is considered. To realize the downlink CSI acquisition, the BS first broadcasts the downlink pilot signal to the UEs, and then each UE estimates the CSI and feeds it back to the BS. We focus on the CE and feedback of a single UE, and its received signal $\mathbf{y}_p \in \mathbb{C}^M$ on the p-th subcarrier in M successive time slots can be expressed as

$$\mathbf{y}_p^{\mathrm{T}} = \mathbf{h}_p^{\mathrm{T}} \mathbf{X} + \mathbf{n}_p^{\mathrm{T}}, \tag{1}$$

where $\mathbf{X} \in \mathbb{C}^{N_{\mathrm{BS}} \times M}$ is the transmit signal in M successive time slots, $\mathbf{h}_p \in \mathbb{C}^{N_{\mathrm{BS}}}$ is the p-th subcarrier's channel, and $\mathbf{n}_p^{\mathrm{T}}$ is complex additive white Gaussian noise (AWGN) with zero mean and covariance matrix $\sigma_n^2 \mathbf{I}$. By stacking the received signals over all the subcarriers, the received signal $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_P]^{\mathrm{T}} \in \mathbb{C}^{P \times M}$ can be written as

$$\mathbf{Y} = \mathbf{H}_s \mathbf{X} + \mathbf{N}, \tag{2}$$

where $\mathbf{H}_s = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_P]^{\mathrm{T}} \in \mathbb{C}^{P \times N_{\mathrm{BS}}}$ is the frequency-spatial domain channel matrix, and $\mathbf{N} = [\mathbf{n}_1, \mathbf{n}_2, \ldots, \mathbf{n}_P]^{\mathrm{T}}$ is the AWGN matrix. Furthermore, we can obtain the frequency-angle domain channel $\mathbf{H}_a \in \mathbb{C}^{P \times N_{\mathrm{BS}}}$ as [2]

$$\mathbf{H}_a = \mathbf{H}_s \mathbf{F}, \tag{3}$$

where $\mathbf{F} \in \mathbb{C}^{N_{\mathrm{BS}} \times N_{\mathrm{BS}}}$ a discrete Fourier transform (DFT) matrix. Due to the angular-domain sparsity, each row of $\mathbf{H}_a$ is a sparse vector. Hence, (2) can be rewritten as

$$\mathbf{Y} = \mathbf{H}_a \mathbf{F}^{\mathrm{H}} \mathbf{X} + \mathbf{N} = \mathbf{H}_a \widetilde{\mathbf{X}} + \mathbf{N}, \tag{4}$$

where $\widetilde{\mathbf{X}} = \mathbf{F}^{\mathrm{H}} \mathbf{X}$.

B. Pilot Design and CSI Acquisition

The structure of the proposed SWTCAN is shown in Fig. 1. We use a linear layer without bias to model the pilot design. The received pilot signal is passed through a linear layer, and its dimension is restored to be the same as the channel. The core of the compressor consists of 8 Swin Transformer blocks [29]. These blocks are responsible for extracting high-dimensional features from the input signal, leveraging a 96-dimensional embedding space. Each pair of Swin Transformer blocks starts with LayerNorm, followed by shifted window multi-head self-attention (SW-MSA) in the first block and window-based multi-Head self-attention (W-MSA) in the second, both paired with multilayer perceptron (MLPs) and residual connections. This structure captures the local and global features of CSI effectively. The features passes

through a linear layer with an activation function and form a codeword. The compressed codeword is then quantized into B bits vector $\mathbf{q}$ by a quantization layer. The entire CSI compressor can be expressed as

$$\mathbf{q} = \mathcal{Q}\left(f_{\downarrow}\left(\mathbf{Y}, \boldsymbol{\theta}_{\downarrow}\right)\right) \in \mathbb{R}^{B}, \tag{5}$$

where $\mathcal{Q}(\cdot)$, $f_{\downarrow}(\cdot, \cdot)$, and $\boldsymbol{\theta}_{\downarrow}$ denote the quantization function, compression function and learnable neural network parameters in the compressor, respectively.

The structure of the CSI reconstructor at the BS is similar to that of the compressor. The received bit vector is transformed by a dequantization layer and a linear layer to change the feature dimension, which is then inputted to Swin Transformer blocks. Finally, the features are up-sampled by a patch expanding layer to reconstruct the downlink CSI $\widehat{\mathbf{H}}_a$ at the BS. The entire CSI reconstructor can be expressed as

$$\widehat{\mathbf{H}}_a = f_{\uparrow}\left(\mathcal{Q}^{-1}\left(\mathbf{q}\right), \boldsymbol{\theta}_{\uparrow}\right), \tag{6}$$

where $f_{\uparrow}(.,.)$, $\mathcal{Q}^{-1}(\cdot)$, and $\boldsymbol{\theta}_{\uparrow}$ denote the reconstruction function, dequantization function and learnable neural network parameters, respectively. By adopting the normalized mean squared error (NMSE) $L_1 = \frac{\|\widehat{\mathbf{H}}_a - \mathbf{H}_a\|_F^2}{\|\mathbf{H}_a\|_F^2}$ as the loss function, we can perform E2E training on the proposed SWTCAN.

III. GENERATIVE PRE-TRAINING AND FEDERATED-TUNING FOR THE PROPOSED SWTCAN

Our proposed CSI-GPT framework integrates the proposed VAE-CSG to pre-train SWTCAN with a small number of CSI samples from the current cell. We also adopt a FL-based online fine-tuning to further improve the performance of pre-trained SWTCAN, which has much lower communication overhead than the CL scheme. The procedure of CSI-GPT with federated-tuning is summarized in Algorithm 1.

A. Generative AI-Based Pre-Training

The proposed VAE-based generative network for generating CSI samples, called VAE-CSG, pre-trains the SWTCAN so that it can initially learn the generalized CSI features before fine-tuning it, which helps the model to perform better in the subsequent task and accelerate the convergence speed. As shown in line 1 of Algorithm 1, the BS initially pre-trains the VAE-CSG using a large amount of CSI samples generated by the channel simulator, and these samples typically have a different channel distribution from the current cell. Then the VAE-CSG is fine-tuned using a limited number of CSI samples from the current cell, which are obtained from the uplink CE at a high signal-to-noise ratio (SNR).[1]

The VAE-CSG comprises an encoder and a decoder, which are jointly trained using an E2E method and only the decoder is utilized for generating CSI samples. Denote the parameters of the encoder and decoder of VAE-CSG as $\boldsymbol{\psi}$ and $\boldsymbol{\omega}$, respectively. The output of the encoder, i.e., the latent variable, is denoted as $\mathbf{z} = f_{enc}(\mathbf{H}_a; \boldsymbol{\psi})$. Similarly, the output of the decoder is denoted as $f_{dec}(\mathbf{z}; \boldsymbol{\omega})$. The loss function of the VAE-CSG consists of two parts. The first part is the reconstruction loss, $\|f_{dec}(\mathbf{z}; \boldsymbol{\omega}) - \mathbf{H}_a\|_F^2$, which measures the closeness of the VAE-CSG's output to the original input. According to [30], the second part measures the difference between the learned distribution of latent variable $\mathbf{z}$ and the predefined prior distribution

---

Algorithm 1: Proposed CSI-GPT with Federated-Tuning.

1:   BS pre-trains VAE-CSG with simulated CSI samples, and fine-tunes VAE-CSG with limited CSI samples obtained in current cell to generate more CSI samples.
2:   BS pre-trains SWTCAN with data generated by VAE-CSG.
3:   Initialize federated-tuning parameters $\mathbf{m}_0 = 0$, $\mathbf{v}_0 = 0$.
4:   **for** $t = 1, 2, \ldots, T$ **do**
5:     BS broadcasts $\tilde{\boldsymbol{\theta}}_t$ to all UEs.
6:     **for** each UE $u \in \mathcal{S}$ **in parallel do**
7:       Initialize local parameters $\tilde{\boldsymbol{\theta}}_t^{u,0} = \tilde{\boldsymbol{\theta}}_t$.
8:       Use SGD for local model updates, according to (9).
9:     **end for**
10:   Perform AirComp, according to (10).
11:   BS updates the model parameter $\tilde{\boldsymbol{\theta}}_{t+1}$, according to (11).
12: **end for**

$p_0(\mathbf{z}) \sim \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$. Let the learned distribution of $\mathbf{z}$ be $p_{\boldsymbol{\psi}}(\mathbf{z}|\mathbf{H}_a)$. Using Kullback-Leibler (KL) divergence, the second part of the loss is $\mathrm{D}_{KL}(p_{\boldsymbol{\psi}}(\mathbf{z}|\mathbf{H}_a)||p_0(\mathbf{z}))$. Hence, the loss function of VAE-CSG is expressed as

$$L_2(\boldsymbol{\psi}, \boldsymbol{\omega}) = \|f_{dec}(\mathbf{z}, \boldsymbol{\omega}) - \mathbf{H}_a\|_F^2$$
$$+ l \cdot \mathrm{D}_{KL}(p_{\boldsymbol{\psi}}(\mathbf{z}|\mathbf{H}_a)||p_0(\mathbf{z})), \tag{7}$$

where l is a predefined hyper-parameter. The second term in (7) enhances the diversity and quality of the generated CSI samples by enforcing the encoder to produce a latent variable that closely matches the standard Gaussian distribution. We use the data generated by VAE-CSG to pre-train SWTCAN (line 2 in Algorithm 1). The performance of pre-trained SWTCAN is suboptimal, since the CSI distributions used in pre-training and testing in practical deployment are usually different.

B. Federated Learning-Based Online Fine-Tuning

To enhance the performance of the pre-trained SWTCAN, fine-tuning is necessary. Adopting the CL strategy for fine-tuning would require the BS to aggregate a large number of CSI samples from UEs in the current cell, resulting in excessive uplink communication overhead. The downlink CSI used for fine-tuning can be obtained by the BS broadcasting the pilot signals, facilitating each UE to estimate its own downlink CSI based on the pilot signals. To address the high communication overhead and privacy issue caused by feeding these CSI samples back to the BS, we utilize the communication-efficient federated-tuning and AirComp to fine-tune the SWTCAN, while reducing uplink communication costs and overhead.

1) Communication-Efficient Federated-Tuning: Uploading the entire parameter set $\boldsymbol{\theta}$ of SWTCAN for fine-tuning would result in prohibitive uplink communication overhead. We opt to freeze the majority parameters of SWTCAN and upload only a minority of the entire parameters, denoted by $\tilde{\boldsymbol{\theta}}$. Hence, in federated-tuning, we solve the optimization:

$$\min_{\tilde{\boldsymbol{\theta}} \in \mathbb{R}^d} L_1(\tilde{\boldsymbol{\theta}}) = \frac{1}{U} \sum_{u=1}^{U} L_1^u(\tilde{\boldsymbol{\theta}}), \tag{8}$$

where $d = |\tilde{\boldsymbol{\theta}}|_c$ is the dimension of the learnable parameters and $L_1^u(\tilde{\boldsymbol{\theta}})$ is the NMSE loss function of the u-th UE.

2) AirComp for Efficient Federated-Tuning: To accelerate federated-tuning convergence and minimize the uplink communication rounds, we employ the federated AMSGrad with max stabilization (FedAMS) [31] in conjunction with AirComp. The BS begins by

---

[1]Although the reciprocity of uplink and downlink channels does not hold in FDD, they usually have similar distributions and features. However, due to the limited transmit power of UEs, the uplink SNR is usually low, and high-quality uplink CSI samples at high SNR are limited.

initializing the SWTCAN model with pre-trained parameters. These parameters have been pre-trained using a large dataset of CSI samples generated by the VAE-CSG. In the t-th communication round, $1 \leq t \leq T$, the BS broadcasts the learnable model parameter $\tilde{\boldsymbol{\theta}}_t$ to all UEs (line 5 in Algorithm 1). Due to the heterogeneous user availability, only a fraction of UEs, denoted as $\mathcal{S}$, participates in the t-th communication round. The u-th UE, $\forall u \in \mathcal{S}$, initializes its parameters as $\tilde{\boldsymbol{\theta}}_t^{u,0} = \tilde{\boldsymbol{\theta}}_t$, and then minimizes its local loss function by conducting K local training epochs with the local learning rate η_l through its local dataset (lines 6–9). In the k-th local training epoch, $1 \leq k \leq K$, learnable parameters $\tilde{\boldsymbol{\theta}}_t^{u,k}$ can be updated by stochastic gradient descent (SGD), which is expressed as

$$\tilde{\boldsymbol{\theta}}_t^{u,k} = \tilde{\boldsymbol{\theta}}_t^{u,k-1} - \eta_l \nabla L_1^u(\tilde{\boldsymbol{\theta}}_t^{u,k-1}), \tag{9}$$

where ∇ denotes the gradient operator. After K local training epochs, instead of sending the entire model back to the BS, the u-th UE sends the model difference $\Delta \tilde{\boldsymbol{\theta}}_t^u = \tilde{\boldsymbol{\theta}}_t^{u,K} - \tilde{\boldsymbol{\theta}}_t^{u,0}$ to the BS, and the BS receives the sum of the local model differences from multiple devices based on AirComp (line 10 in Algorithm 1), which can be expressed as

$$\boldsymbol{\delta}_t = \frac{1}{|\mathcal{S}|_c} \sum_{u \in \mathcal{S}} \Delta \tilde{\boldsymbol{\theta}}_t^u + \mathbf{n}^+, \tag{10}$$

where $\mathbf{n}^+ \in \mathbb{R}^d$ is the noise in the uplink aggregation process, $\boldsymbol{\delta}_t$ represents the noisy model difference, which is also treated as a pseudo gradient to update the model at the BS. According to [31], the BS updates the learnable parameters $\tilde{\boldsymbol{\theta}}_{t+1}$ for the $(t+1)$-th round (line 11 in Algorithm 1) as

$$\tilde{\boldsymbol{\theta}}_{t+1} = \tilde{\boldsymbol{\theta}}_t + \eta \frac{\mathbf{m}_t}{\sqrt{\mathbf{v}_t}}, \tag{11}$$

where η is the global learning rate, $\mathbf{m}_t$ is the momentum and $\mathbf{v}_t$ is the variance in the t-th round, which are updated by $\mathbf{m}_t = \beta_1 \mathbf{m}_{t-1} + (1-\beta_1)\boldsymbol{\delta}_t$ and $\mathbf{v}_t = \max\{\mathbf{v}_{t-1} + (1-\beta_2)\boldsymbol{\delta}_t^2, \mathbf{v}_{t-1}\}$ with the hyperparameters β_1 and β_2.

IV. SIMULATION RESULTS

We use the Sionna library in Python to generate MIMO channel realizations. The training and test datasets contain the same number of various types of CDL channels, namely, CDL-A, CDL-B, and CDL-C, where the CDL channel models are adopted from 3GPP standards [20]. The sizes of the training, validation, and test channel datasets are 6000, 2000, and 2000, respectively. The carrier frequency is 28 GHz, the number of subcarriers is $P = 256$ and the subcarrier spacing is 240 kHz. The BS is equipped with the UPA of $N_{\text{BS}} = 16 \times 16 = 256$ antennas, with antenna space $\frac{\lambda_c}{2}$, where λ_c is the signal wavelength. The delay spread is 30 ns, and the downlink SNR is 20 dB.

A. Performance of Proposed SWTCAN

We compare the proposed SWTCAN with the following baselines. **Baseline 1**: The Transformer-based network for CE and CSI feedback [18], denoted as 'Transformer CE + Transformer Feedback'. **Baseline 2**: The multiple-measurement-vectors (MMV)-learned approximate message passing (LAMP) algorithm [32] for CE and the bit-level CsiNet scheme with an attention mechanism [33] or
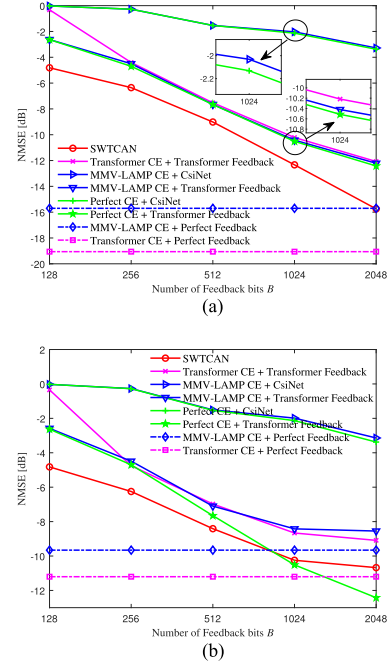

Fig. 2. NMSE performance of different schemes versus the feedback overhead B for CDL-B. (a) $\rho = 8$. (b) $\rho = 16$.

a Transformer-based network for CSI feedback, denoted as 'MMV-LAMP CE + CsiNet/Transformer Feedback'. **Baseline 3**: The perfect downlink channel estimate and the bit-level CsiNet scheme with an attention mechanism [33] or a Transformer-based network for CSI feedback, denoted as 'Perfect CE + CsiNet/Transformer Feedback'. **Baseline 4**: The MMV-LAMP algorithm or a Transformer-based network for CE and the perfect CSI feedback to the BS, denoted as 'MMV-LAMP/Transformer CE + Perfect Feedback'.

Fig. 2 shows the NMSE performance achieved by different schemes versus the feedback overhead B. Both the training and test datasets are CDL-B channels. Fig. 2(a) indicates that at a compression ratio $\rho = \frac{N_{\text{BS}}}{M} = 8$, the main factor influencing the performance is the CSI feedback scheme. Our SWTCAN demonstrates excellent performance across all feedback bit numbers B, outperforming the baseline schemes and approaching the performance of perfect CSI feedback at $B = 2048$. Fig. 2(b) shows that at a higher $\rho = 16$, the CE algorithm also affects the final performance. Simulations on CDL-A and CDL-C channels, not showing due to space limits, also verify the same advantages of our proposed SWTCAN.

In addition, we have analyzed the complexity of the proposed algorithm and the baselines. The complexity of the proposed SWTCAN mainly comes from W-MSA or SW-MSA layers, i.e., $\mathcal{O}(LPN_{\text{BS}}C^2 + LW^2 PN_{\text{BS}}C) \approx 1.4 \times 10^7$, where L is the number of layers in NN, C is the number of channels of CSI, and W is the size of windows. The complexity of Transformer mainly comes from self-attention layers, i.e., $\mathcal{O}(LP^2 d_{model}) \approx 1.5 \times 10^8$, where d_{model} is the dimension of linear embedding in Transformer. The complexity of the MMV-LAMP algorithm mainly comes from matrix multiplication operations, i.e., $\mathcal{O}(GMPN_{\text{BS}}^2 I) \approx 1.1 \times 10^{10}$, where G is the oversampling factor in redundant dictionary, I is the number of iterations. The complexity of CsiNet mainly comes from convolutional layers, i.e., $\mathcal{O}(PN_{\text{BS}}N_{co}^2 \sum_{i=1}^{L} n_{i-1} n_i) \approx 2.4 \times 10^6$, where N_{co} is the size of the convolutional filters, n_{i-1} and n_i are the numbers of input and output feature maps of the i-th convolutional layer. The values of the above parameters can be obtained from the open source code. It is evident that

TABLE I
PRE-TRAINING TEST NMSE (dB) PERFORMANCE COMPARISON OF DIFFERENT
PRE-TRAINING SCHEMES UNDER THE CDL-B CHANNEL

Number of feedback bits	Scheme A	Scheme B	Scheme C	Scheme D	Proposed
B=512	0.2009	-2.8306	-3.8199	-4.0628	**-5.0361**
B=1024	0.2491	-3.7469	-4.8259	-5.5036	**-7.3138**
B=2048	-0.1258	-3.9417	-5.5449	-6.6860	**-9.3678**

the proposed SWTCAN exhibits lower complexity than the baselines while maintaining good performance.

B. Performance of Proposed Generative Pre-Training

We evaluate the performance of generative pre-training on SWTCAN at $\rho=8$ and $B=512$, 1024 and 2048 bits with the value of l in (7) set to 0.00025. We assume that the BS has 6000 CDL-A CSI samples, which are obtained from the channel model generator. But the true CSI distribution in the BS's cell follows a different CDL-B distribution. The BS only has 120 high-quality CDL-B samples, which are obtained from the uplink CE. We compare the proposed scheme and three benchmark schemes for ablation study. **Scheme A**: We pre-train SWTCAN with 6000 CDL-A samples. **Scheme B**: We pre-train SWTCAN with 120 CDL-B samples. **Scheme C**: We pre-train SWTCAN with 6000 CDL-A samples, fine-tune it with 120 CDL-B samples. **Scheme D**: We train VAE-CSG with 120 CDL-B samples, and then use it to generate 6000 CSI samples to pre-train SWTCAN. By contrast, in the **Proposed** scheme, we pre-train VAE-CSG with 6000 CDL-A samples, fine-tune it with 120 CDL-B samples, and finally generate 6000 CSI samples to pre-train SWTCAN. These four pre-training schemes are tested on 2000 CDL-B samples, and the pre-training test NMSEs are compared in Table I. It can be seen that the proposed VAE-CSG outperforms the other three benchmark schemes, demonstrating its effectiveness. Based on the proposed pre-training strategy, the performance of SWTCAN at $\rho=8$ and $B=2048$ bits reaches the NMSE of -9.3678 dB, which is still a bit short of -15.7 dB shown in Fig. 2(a). Therefore, we use federated-tuning to further improve the performance.

C. Performance of Proposed Federated-Tuning Method

In this ablation study, there are $U=600$ UEs, each having $N_{\text{FL}}^s = 10$ actual CSI samples. During each communication round of federated-tuning, 10% of UEs, i.e., 60 UEs, are involved, and each UE conducts $K=2$ local training epochs with the local training rate $\eta_l=0.001$ to facilitate online fine-tuning of SWTCAN at $\rho=8$ and $B=2048$ bits. The hyperparameters in (11) are set to $\eta=1$, $\beta_1=0.9$, and $\beta_2=0.99$. We freeze the parameters of the SWTCAN except for the last two layers [2] in the decoder. SNR is set to 20 dB for both downlink CE and uplink AirComp. For federated-tuning, computation (updating the trainable parameters $\tilde{\boldsymbol{\theta}}$) is done in the UE, and the model updates of multiple UEs are transmitted using AirComp[3]. For CL, the BS collects CSI samples via orthogonal transmission, and then updates the whole model parameters $\boldsymbol{\theta}$. Note that $|\tilde{\boldsymbol{\theta}}|_c=3617280$, $|\boldsymbol{\theta}|_c=32623524$, hence $|\tilde{\boldsymbol{\theta}}|_c \approx 0.11|\boldsymbol{\theta}|_c$.

---

[2] Our study showed that freezing the last two layers of all Swin Transformer Blocks (with 3,617,280 unfrozen parameters) resulted in a final performance of -10.722 dB after 25 global epochs. Freezing only the last layer (2,509,299 unfrozen parameters) led to a performance of -10.299 dB, while freezing half of the last layer (2,471,667 unfrozen parameters) achieved -10.202 dB. The chosen freezing scheme strikes a balance between communication efficiency and performance, which is verified in the simulations.

[3] Note that the uplink communication overhead can be further reduced using gradient compression methods [23]. Due to space limitation, this point is not discussed in this paper, which will be investigated in future.

TABLE II
CHARACTERIZATION OF COMMUNICATION OVERHEAD, COMPUTATION COST
AND COMPUTATION SPEED OF FEDERATED-TUNING AND CL SCHEMES

Schemes / Items	Federated-tuning (one global epoch)	CL (one CSI sample)
Communication overhead ($\times 10^5$ real numbers)	$\|\tilde{\boldsymbol{\theta}}\|_c \approx 36$	$2PN_{\text{BS}} \approx 1.3$
Computation cost ($\times 10^9$ FLOPs) [3]	$\zeta_{\text{UE}} \approx 2.6$	$\zeta_{\text{BS}} \approx 3.4$
Computational speed (FLOPs/s)	κ_{UE}	$\kappa_{\text{BS}} = \gamma\kappa_{\text{UE}}$

[3] The computation cost containing both forward and backward propagations is numerically calculated using torch-summary [34]. Note that federated-tuning and CL have the same forward propagation, while federated-tuning has less computation cost in the backward propagation.
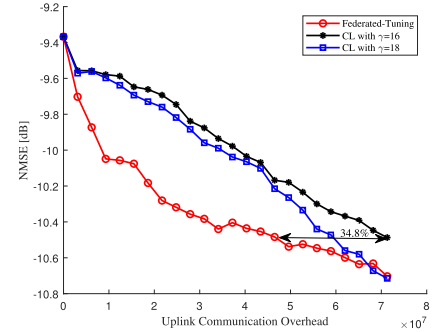

Fig. 3. Performance comparison of federated-tuning and CL schemes versus the uplink communication overhead.

Table II characterizes the communication overhead, computation cost and computation speed of the federated-tuning (one global epoch) and the CL scheme (one CSI sample). For federated-tuning, communication overhead during each global epoch is denoted as $|\tilde{\boldsymbol{\theta}}|_c$, representing the number of trainable parameters of SWTCAN. While, the communication overhead of CL is calculated as $2PN_{\text{BS}}$ for each CSI sample. The computation cost is measured in FLOPs using the torch-summary tool, where ζ_{UE} and ζ_{BS} represent the computation cost in federated-tuning and CL, respectively. Computation speed, in FLOPs/s, is indicated by κ_{BS} and κ_{UE} for the BS and UE, respectively, where their ratio $\gamma = \frac{\kappa_{\text{BS}}}{\kappa_{\text{UE}}}$ reflects the computational power difference between the BS and UEs.

For a fair comparison, later we will compare the NMSE performance of the two schemes, given the same communication resources and the same computation time. Here, we first calculate how many CSI samples the BS can collect in CL, given a fixed communication resource. Specifically, consider T_0 global epochs of federated-tuning, within the communication overhead is $T_0|\tilde{\boldsymbol{\theta}}|_c$. Using the same communication resources, the BS in CL can collect $N_{\text{CL}}^s = \frac{T_0|\tilde{\boldsymbol{\theta}}|_c}{2PN_{\text{BS}}}$ CSI samples for central training. Furthermore, we calculate how many training epochs CL can have, given a fixed computation time. In particular, given the number of global epochs T_0, the computation cost ζ_{UE}, the number of local CSI samples N_{FL}^s, the number of local training epochs K, and the computation speed κ_{UE}, the computation time of federated-tuning can be calculated as $\tau = \frac{T_0 N_{\text{FL}}^s K \zeta_{\text{UE}}}{\kappa_{\text{UE}}}$. Within the same computation time τ in CL, given the computation cost ζ_{BS}, the number of CSI samples collected in the BS N_{CL}^s, and the computation speed in the BS κ_{BS}, we can obtain the number of training epochs of CL as $K_{\text{CL}} = \frac{\tau\kappa_{\text{BS}}}{N_{\text{CL}}^s \zeta_{\text{BS}}}$. Although the BS is typically computationally more powerful than a UE, i.e., $\gamma > 1$, our proposed federated-tuning method still shows advantages due to the collaboration of multiple UEs.

Fig. 3 compares the NMSE of the proposed federated-tuning with that of the CL scheme versus uplink communication overhead. It can

be seen that our federated-tuning method exhibits superior NMSE performance compared to the CL scheme under the same computation time and the same uplink communication overhead. Specifically, to achieve an equivalent NMSE performance in the same computation time, compared to the CL scheme with $\gamma = 16$ (this situation is possible, e.g., a Nvidia 3090Ti graphics card with a computing power of 41.6 TFLOPs on the BS and an iPad Air with a computing power of 2.6 TFLOPs on the M1 chip of the UE), our proposed algorithm reduces uplink communication overhead by up to 34.8%. For the CL scheme, as the ratio of computational speed γ increases, the number of training epochs that BS can conduct also increase given the same computation time, resulting in improved performance.

V. CONCLUSION

We have proposed a Swin Transformer-based CSI acquisition network called SWTCAN to jointly design the pilot, CSI compression and CSI reconstruction. In order to solve the training data scarcity problem as the actual CSI samples are difficult to measure, we have designed the VAE-CSG to generate CSI samples for pre-training SWTCAN. The combination of VAE-CSG and SWTCAN constitutes the downlink CSI acquisition network based on a generative pre-trained Transformer at the BS. To further enhance the performance of the pre-trained SWT-CAN, we have utilized the communication-efficient federated-tuning and AirComp to fine-tune the SWTCAN, while substantially reducing uplink communication overhead. Simulations have demonstrated that our proposed SWTCAN has better performance compared to the state-of-the-art schemes, and have verified the communication efficiency of the proposed federated-tuning method. However, DL methods still impose high complexity and memory requirements on UEs and the implementation of AirComp introduces practical issues that require future efforts.

REFERENCES

[1] H. Liu et al., "Near-space communications: The last piece of 6G space-air-ground-sea integrated network puzzle," *Space Sci. Technol.*, vol. 4, 2024, Art. no. 0176, doi: 10.34133/space.0176.

[2] Z. Gao, L. Dai, Z. Wang, and S. Chen, "Spatially common sparsity based adaptive channel estimation and feedback for FDD massive MIMO," *IEEE Trans. Signal Process.*, vol. 63, no. 23, pp. 6169–6183, Dec. 2015.

[3] B. Jiang et al., "Total and minimum energy efficiency tradeoff in robust multigroup multicast satellite communications," *Space Sci. Technol.*, vol. 3, 2023, Art. no. 0059, doi: 10.34133/space.0059.

[4] R. Zhang et al., "Integrated sensing and communication with massive MIMO: A unified tensor approach for channel and target parameter estimation," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 8571–8587, Aug. 2024, doi: 10.1109/TWC.2024.3351856.

[5] J. Xu, L. You, G. C. Alexandropoulos, X. Yi, W. Wang, and X. Gao, "Near-field wideband extremely large-scale MIMO transmissions with holographic metasurface-based antenna arrays," *IEEE Trans. Wireless Commun.*, vol. 23, no. 9, pp. 12054–12067, Sep. 2024, doi: 10.1109/TWC.2024.3387709.

[6] P. Zhu, H. Lin, J. Li, D. Wang, and X. You, "High-performance channel estimation for mmWave wideband systems with hybrid structures," *IEEE Trans. Commun.*, vol. 71, no. 4, pp. 2503–2516, Apr. 2023.

[7] J. Fang, P. Zhu, J. Li, F.-C. Zheng, and X. You, "Cell-free mMIMO systems in short packet transmission regime: Pilot and power allocation," *IEEE Trans. Veh. Technol.*, vol. 73, no. 6, pp. 8322–8337, Jun. 2024.

[8] Y. Ye, L. You, J. Wang, H. Xu, K.-K. Wong, and X. Gao, "Fluid antenna-assisted MIMO transmission exploiting statistical CSI," *IEEE Commun. Lett.*, vol. 28, no. 1, pp. 223–227, Jan. 2024.

[9] X. Ma and Z. Gao, "Data-driven deep learning to design pilot and channel estimator for massive MIMO," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5677–5682, May 2020.

[10] M. B. Mashhadi and D. Gündüz, "Pruning the pilots: Deep learning-based pilot design and channel estimation for MIMO-OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 20, no. 10, pp. 6315–6328, Oct. 2021.

[11] C.-K. Wen, W.-T. Shih, and S. Jin, "Deep learning for massive MIMO CSI feedback," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 748–751, Oct. 2018.

[12] J. Guo, C.-K. Wen, S. Jin, and G. Y. Li, "Convolutional neural network-based multiple-rate compressive sensing for massive MIMO CSI feedback: Design, simulation, and analysis," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2827–2840, Apr. 2020.

[13] M. Nerini, V. Rizzello, M. Joham, W. Utschick, and B. Clerckx, "Machine learning-based CSI feedback with variable length in FDD massive MIMO," *IEEE Trans. Wireless Commun.*, vol. 22, no. 5, pp. 2886–2900, May 2023.

[14] Y. Cui et al., "Lightweight neural network with knowledge distillation for CSI feedback," *IEEE Trans. Commun.*, vol. 72, no. 8, pp. 4917–4929, Aug. 2024.

[15] J. Guo, C.-K. Wen, S. Jin, and G. Y. Li, "Overview of deep learning-based CSI feedback in massive MIMO systems," *IEEE Trans. Commun.*, vol. 70, no. 12, pp. 8017–8045, Dec. 2022.

[16] J. Guo et al., "Deep learning for joint channel estimation and feedback in massive MIMO systems," *Digit. Commun. Netw.*, vol. 10, no. 1, pp. 83–93, 2024.

[17] J. Guo, C.-K. Wen, and S. Jin, "CANet: Uplink-aided downlink channel acquisition in FDD massive MIMO using deep learning," *IEEE Trans. Commun.*, vol. 70, no. 1, pp. 199–214, Jan. 2022.

[18] Y. Wang, Z. Gao, D. Zheng, S. Chen, D. Gündüz, and H. V. Poor, "Transformer-empowered 6G intelligent networks: From massive MIMO processing to semantic communication," *IEEE Wireless Commun.*, vol. 30, no. 6, pp. 127–135, Dec. 2023.

[19] K. Han et al., "A survey on vision transformer," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 87–110, Jan. 2023.

[20] 3GPP, "Study on channel model for frequencies from 0.5 to 100 GHz," 3rd Generation Partnership Project (3GPP), Sophia Antipolis, France, Tech. Rep. TR 38.901, May 2017.

[21] H. Xiao, W. Tian, W. Liu, and J. Shen, "ChannelGAN: Deep learning-based channel modeling and generating," *IEEE Wireless Commun. Lett.*, vol. 11, no. 3, pp. 650–654, Mar. 2022.

[22] Z. Yang et al., "Federated learning for 6G: Applications, challenges, and opportunities," *Engineering*, vol. 8, pp. 33–41, 2022.

[23] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Trans., Wireless Commun.*, vol. 19, no. 5, pp. 3546–3557, May 2020.

[24] L. Qiao, Z. Gao, M. B. Mashhadi, and D. Güundüz, "Massive digital over-the-air computation for communication-efficient federated edge learning," *IEEE J. Sel. Areas Commun.*, vol. 42, no. 11, pp. 3078–3094, Nov. 2024.

[25] J. Chen et al., "FedTune: A deep dive into efficient federated fine-tuning with pre-trained transformers," 2022, *arXiv:2211.08025*.

[26] A. M. Elbir and S. Coleri, "Federated learning for channel estimation in conventional and RIS-assisted massive MIMO," *IEEE Trans. Wireless Commun.*, vol. 21, no. 6, pp. 4255–4268, Jun. 2022.

[27] L. Dai and X. Wei, "Distributed machine learning based downlink channel estimation for RIS assisted wireless communications," *IEEE Trans. Commun.*, vol. 70, no. 7, pp. 4900–4909, Jul. 2022.

[28] L. Zhao, H. Xu, Z. Wang, X. Chen, and A. Zhou, "Joint channel estimation and feedback for mm-wave system using federated learning," *IEEE Commun. Lett.*, vol. 26, no. 8, pp. 1819–1823, Aug. 2022.

[29] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 11–17, 2021, pp. 10012–10022.

[30] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. Int. Conf. Learn. Representations*, 2014.

[31] Y. Wang, L. Lin, and J. Chen, "Communication-efficient adaptive federated learning," in *Proc. Int. Conf. Mach. Learn.*, Jul. 17–23, 2022, pp. 22802–22838.

[32] X. Ma, Z. Gao, F. Gao, and M. Di Renzo, "Model-driven deep learning based channel estimation and feedback for millimeter-wave massive hybrid MIMO systems," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2388–2406, Aug. 2021.

[33] C. Lu, W. Xu, S. Jin, and K. Wang, "Bit-level optimized neural network for multi-antenna channel quantization," *IEEE Wireless Commun. Lett.*, vol. 9, no. 1, pp. 87–90, Jan. 2020.

[34] T. Yep, "Torch-summary 1.4.5," Accessed on: Dec. 24, 2020. [Online] Available: https://pypi.org/project/torch-summary/