

# 딥러닝 기반 CSI 압축 모델의 Fine-Tuning 성능 분석

한송아, 홍송남\*  
한양대학교

hsssonga24@hanyang.ac.kr, \*snhong@hanyang.ac.kr

## Performance Analysis of Fine-Tuning for Deep Learning-based CSI Compression

Han Song Ah, Hong Song Nam\*  
Hanyang Univ.

### 요약

대규모 MIMO 시스템에서 채널 상태 정보 (Channel State Information, CSI) 압축 및 피드백은 필수적이며, 딥러닝 기반 CSI 압축 모델은 일반적으로 우수한 성능을 보인다. 그러나 채널 환경이 변화하면 모델의 성능은 저하될 수 있다. 이를 완화하기 위해 사전 학습된 신경망 기반 CSI 압축 모델에 Fine-Tuning 기법을 적용한다. 본 연구는 Encoder-Only와 Full-Model Fine-Tuning 방식을 비교하고, 업데이트 주기에 따른 Rate-Distortion(RD) 성능과 피드백 오버헤드 간의 trade-off를 분석하였다. 실험 결과, Full-Model Fine-Tuning이 일관적으로 우수한 RD 성능을 나타냈으며, 최적 업데이트 전략의 필요성을 확인하였다.

### I. 서론

대규모 MIMO(Multi-Input Multi-Output) 시스템은 다수 안테나를 통해 고용량 전송을 가능하게 하지만, 다운링크 CSI 피드백 과정에서 발생하는 오버헤드가 시스템 성능을 제한한다. 이를 해결하기 위해 최근 딥러닝 기반의 CSI 압축 기법이 활발히 연구되고 있으며, 기존 압축 방식 대비 높은 효율을 보이고 있다 [1][2]. 하지만 이러한 모델 역시 채널 환경이 변화하여 데이터 분포가 달라질 경우, 성능이 저하되는 한계를 가진다.

본 연구에서는 이러한 문제를 완화하기 위해 사전 학습된 모델에 Fine-Tuning 기법을 적용하여 Encoder-Only와 Full-Model 방식을 비교하고 업데이트 주기별 RD 성능과 피드백 오버헤드 간의 trade-off를 분석하여 최적의 업데이트 전략의 필요성을 제시하였다.

### II. CSI 압축 모델

본 연구는 FDD Massive MIMO 환경을 기반으로 한다. 기지국의 안테나 수를  $N_t$ , 서브캐리어 수를  $N_c$ 라 할 때, 다운링크 채널의 수신 신호와 CSI 행렬은 다음과 같이 정의된다.

$$\mathbf{y}_m = \mathbf{h}_m^T \mathbf{w}_m x_m + n_m, \mathbf{H} = [\mathbf{h}_1, \dots, \mathbf{h}_{N_c}] \in \mathcal{C}^{N_t \times N_c} \quad (1)$$

CSI 압축은 [3]에서 제안된 인코더, 디코더, 엔트로피 부호화기로 구성된 신경망 기반 모델  $c = (f_\phi, g_\theta, \gamma_\theta)$ 를 사용한다. 인코더  $f_\phi$ 는 CSI를 잠재 표현으로 변환 후 양자화하여 비트스트림을 생성하고, 디코더  $g_\theta$ 는 이를 복원한다. 학습 시 양자화는 균일 노이즈  $\Delta Z \sim \mathcal{U}(0, 1)$ 를

더해 근사하였으며, 잠재 변수 분포  $p_\theta$ 에는 DeepFactorized [4]를 적용하였다 (그림 1).

손실 함수는 RD 성능을 고려하여 압축률(Rate)와 왜곡(Distortion)의 가중합으로 정의된다.

$$L_{RD}(\phi, \theta) = R(c) + \lambda D(c) \quad (2)$$

성능 검증에는 DeepMIMO ‘O1’ 데이터셋을 사용하였다.  $32 \times 32$ 와  $64 \times 64$  크기에서 각각 40,000개와 28,000개의 샘플을 사용하였고, 학습, 검증, 테스트 비율은 각각 40%, 40%, 20%로 설정하였다. Adam (학습률 0.001)을 이용해 200 epoch 학습을 수행했으며, 손실 함수의 가중치는  $\lambda = [5 \times 10^4, 10^5, 5 \times 10^5, 10^6]$ 를 적용하였다. 시뮬레이션 결과(그림 2),  $\lambda$ 값이 증가할수록 Rate가 증가하고 NMSE가 감소하였다.

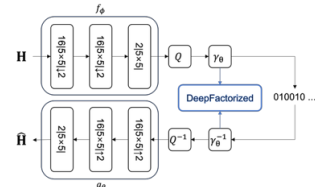


그림 1. CSI 압축 모델 구조

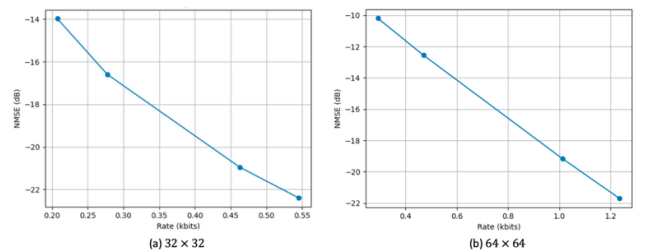


그림 2.  $\lambda$ 값 변화에 따른 Rate-Distortion 성능

### Algorithm 1 Full-model Fine-tuning

**Input:** Global model parameters  $\{\theta_0, \phi_0\}$  trained on a generic dataset, batch size  $B$ , CSI samples from a shifted distribution  $\mathcal{H} = \{\mathcal{H}_T, \mathcal{H}_E\}$ , model prior  $p[\delta]$

**Output:** CSI matrix  $\hat{\mathbf{H}}$

```

1: Initialize model parameters:  $\phi = \phi_0, \theta = \theta_0$ 
2: for epoch = 1 to num.epochs do
3:   for each batch  $b$  in  $\mathcal{H}_T$  do
4:     Load batch  $\mathbf{H} \in \{\mathcal{H}_b\}_{b=1}^B$  from  $\mathcal{H}_T$ 
5:     Quantize updated decoder parameters:  $\bar{\theta} = Q_t(\delta) + \theta_0$ , with  $\delta = \theta - \theta_0$ 
6:     Apply feature encoder and quantization:  $\mathbf{Z} = f_\phi(\mathbf{H})$ ,  $\tilde{\mathbf{Z}} = \mathbf{Z} + \Delta\mathbf{Z}$ 
7:     Apply feature decoder:  $\hat{\mathbf{H}} = g_{\bar{\theta}}(\tilde{\mathbf{Z}})$ 
8:     Compute loss  $L_{RD}(\phi, \bar{\theta})$ 
9:     Backpropagate using STE for  $Q_t(\delta)$  and update  $\theta, \phi$ 
10:   end for
11: end for
12: return Fine-tuned model parameters  $\{\phi^*, \theta^*\}$ 
13: Compress  $\mathbf{H} \in \mathcal{H}_E$  to latent representation  $\tilde{\mathbf{Z}} = Q(f_{\phi^*}(\mathbf{H}))$ 
14: Compute quantized model parameters:  $\bar{\theta} = \theta_0 + \delta$ , with  $\delta = Q_t(\theta^* - \theta_0)$ 
15: Entropy encode:  $b_\delta = \gamma(\delta; p[\delta])$ ,  $b_z = \gamma_{\bar{\theta}}(\tilde{\mathbf{Z}}; p_{\bar{\theta}})$ 
16: Entropy decode:  $\hat{\delta} = \gamma^{-1}(b_\delta; p[\delta])$ 
17: Compute updated decoder parameters:  $\bar{\theta} = \theta_0 + \hat{\delta}$ 
18: Entropy decode latent:  $\hat{\mathbf{Z}} = \gamma_{\bar{\theta}}^{-1}(b_z; p_{\bar{\theta}})$ 
19: Apply de-quantization and feature decoder:  $\hat{\mathbf{H}} = Q^{-1}(g_{\bar{\theta}}(\hat{\mathbf{Z}}))$ 

```

알고리즘 1. Full-Model Fine-Tuning

### III. Fine-Tuning

채널 환경 변화로 데이터 분포가 달라질 경우, 사전 학습된 모델의 성능은 저하된다. 이를 완화하기 위해 새로운 환경에서 Encoder-Only(EO)와 Full-Model(FM) Fine-Tuning 방식을 적용하였다.

EO는 인코더만 업데이트하는 반면, FM은 인코더와 디코더, 엔트로피 모델 전체를 업데이트한다. FM에서는 오버헤드를 줄이기 위해 업데이트된 디코더 파라미터  $\delta$ 를 양자화 및 부호화 후 전송하며, 학습 시 Straight-Through Estimator(STE)를 적용한다 (알고리즘 1).  $\delta$ 의 분포로는 Uniform Prior, Truncated Gaussian Prior를 고려하였다.

### IV. 성능 평가

성능 평가는 DeepMIMO의 'O2 Dynamic' 데이터셋에서 200개의 소량 샘플을 사용하여 수행하였다. EO는 200 epoch, FM은 500 epoch 동안 Adam (학습률 0.0005)으로 학습하였다.

비교 대상으로는 디코더가 완전한 업데이트 정보를 오버헤드 없이 보유한다고 가정한 Genie-Aided (GA) 방식 [5]과 입력과 출력을 보정하는 추가 모듈을 부착하여 조정하는 방식인 Translation Module (TM) 방식 [6]을 포함하였다.

그림 3의 결과, EO 및 TM 방식은 성능 개선이 제한적이었으나 FM 방식은 다양한  $\lambda$  값에서 일관되게 우수한 RD 성능을 나타냈다. Uniform Prior와 Truncated Gaussian Prior 간 RD 성능 차이는 크지 않았다. 이는 두 분포 모델이 디코더 업데이트를 효과적으로 근사함을 보여준다.

또한 업데이트 주기(50, 100, 200 샘플)를 달리하여 FM의 성능-오버헤드 trade-off를 분석하였다 (그림 4). 짧은 주기는 성능 향상은 크지만 피드백 오버헤드가 증가하고, 긴 주기는 오버헤드는 감소하나 적응성은 떨어졌다. 따라서 동적 환경에서는 변화 정도에 맞는 최적의 업데이트 전략이 필요함을 확인하였다.

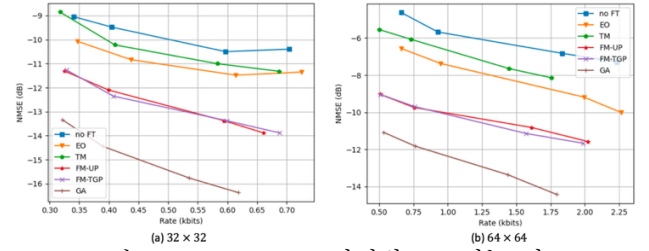


그림 3. Fine-Tuning 방식별 RD 성능 비교

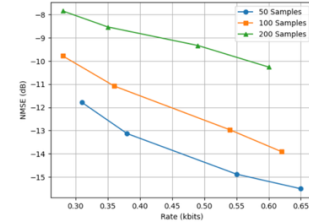


그림 4. 업데이트 주기별 Full-Model Fine-Tuning 성능

### V. 결론

본 연구에서는 딥러닝 기반 CSI 압축 모델이 채널 환경 변화 시 성능이 저하되는 한계를 다루고, 이를 완화하기 위한 Encoder Only와 Full-Model Fine-Tuning 기법을 분석하였다.

결과적으로 FM 방식이 EO 방식보다 RD 성능에서 일관적으로 우수하며, 업데이트 주기 최적화가 성능과 통신 오버헤드 균형에 중요함을 확인하였다.

### 참고 문헌

- [1] Wen, Chao-Kai, et al. "Deep learning for massive MIMO CSI feedback." *IEEE Wireless Commun. Lett.* 7.5 (2018): 748-751.
- [2] Tang, Shunpu, et al. "Dilated convolution based CSI feedback compression for massive MIMO systems." *IEEE Trans. on Veh. Technol.* 71.10 (2022): 11216-11221.
- [3] Sattari, Mehdi, et al. "Dynamic Model Fine-Tuning For Extreme MIMO CSI Compression." *arXiv:2501.18250* (2025).
- [4] Ballé, Johannes, et al. "Variational image compression with a scale hyperprior." *arXiv:1802.01436* (2018).
- [5] Zeng, Jun, et al. "Downlink CSI feedback algorithm with deep transfer learning for FDD massive MIMO systems." *IEEE Trans. Cogn. Commun. Netw.* 7.4 (2021): 1253-1265.
- [6] Liu, Zhenyu, et al. "Deep learning for efficient csi feedback in massive mimo: Adapting to new environments and small datasets." *IEEE Trans. Wireless Commun.* 23.9 (2024): 12297-12312.