Trade-off Between Accuracy and Resources in CSI Feedback Compression

Your Name

I. Trade-off Between Accuracy and Resources

CNNs have demonstrated superior performance in CSI compression tasks compared to compressive sensing methods. While numerous architectures have been proposed, deploying these models on mobile hardware presents significant challenges. Consequently, it becomes imperative to strike a balance between accuracy and resource constraints when considering the practical application of CSI feedback compression on user equipment with limited resources.

Most research in CSI feedback compression focuses on various designs of auto-encoders. A significant portion of the parameters and FLOPs in auto-encoder designs is attributed to the fully connected layer. Considering the CsiNet [1] architecture, initial measurements indicate a computational load of 5.6 million FLOPs and 2.1 million trainable parameters. Upon selective removal of the fully connected layers within the architecture, these metrics were reassessed, revealing a notable reduction. Specifically, the FLOPs decreased to 3.5 million, while the number of trainable parameters plummeted to 3400.

This analysis underscores a substantial optimization opportunity through the elimination of fully connected layers. Notably, the removal of these layers resulted in a reduction of approximately 37.5% in FLOPs, indicating a significant computational burden alleviation. Moreover, the parameter count experienced a staggering decrease of over 99.8%, illuminating the dominant role played by fully connected layers in parameter-heavy neural network architectures such as CsiNet [1].

Additionally, kernel size in CNNs structure stands out as a crucial factor influencing the computational complexity of CNNs within the CsiNet [1] architecture. Initially set at (3,3), altering the kernel size to (7,7) prompts a significant shift in computational requirements. Specifically, this adjustment results in a notable increase, with FLOPs soaring to 20.3 million and the number of parameters remaining at 2.1 million.

This shift underscores the profound impact of kernel size on computational demands within CNNs architectures. Notably, enlarging the kernel size to (7,7) yields a substantial rise in FLOPs, escalating by approximately 362%. Furthermore, the adjustment marginally affects the parameter count, increasing it by a modest 1.44%.

These findings underscore the intricate interplay between kernel size and computational complexity in CNNs architectures such as CsiNet [1]. Such insights are valuable for optimizing network design to achieve desired performance outcomes while managing computational resources effectively.

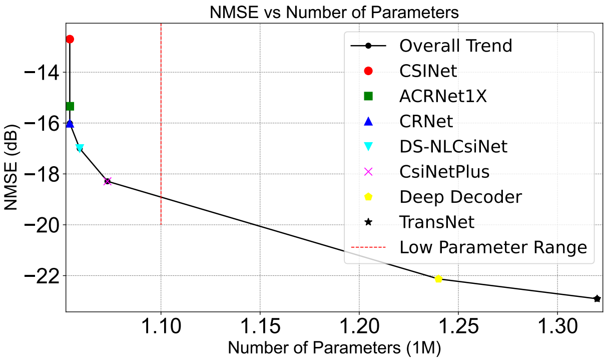

Fig. 1: NMSE vs Number of Parameters

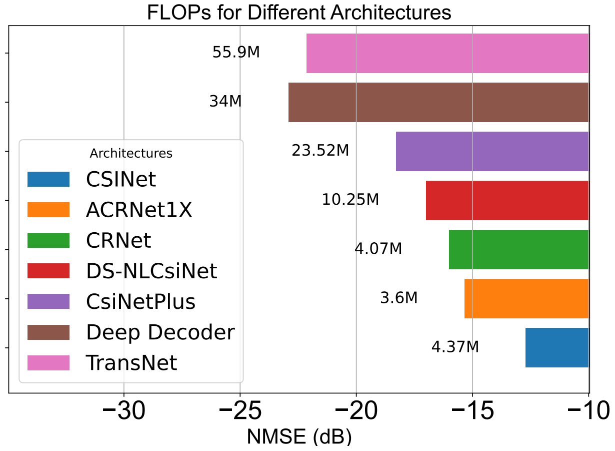

Fig. 2: NMSE vs FLOPs

CsiNet [1] achieves a parameter count under 1 million, while ACRNet [2] and CRNet [3] achieve superior Normalized Mean Square Error (NMSE) performance while maintaining a parameter count similar to CsiNet [1]. DS-NLCsiNet [4] achieves an NMSE near -17 dB with nearly 10 million FLOPs and a parameter count similar to CsiNet. In contrast, Deep Decoder [5] and TransNet [6] have over 50 million FLOPs and 25% and 30% more parameters, respectively, compared to CsiNet (Figure 1 and 2).

Many researchers primarily focus on increasing accuracy without considering the practical deployment of their proposed architectures. However, beyond addressing the imperfect channel, a challenge for evaluating the real performance of deep learning versus compressive sensing methods lies in considering storage constraints. Lu et al. [7] discuss the flexible deployment of their proposed architecture by adjusting kernel sizes for the convolutional neural network part of the auto-encoder and implementing network binarization for the fully connected layer. Lu et al. [7] also propose feature quantization for bitstream generation, replacing the feedback of floating-point numbers, which may not be feasible in digital communication systems.

As highlighted by Lu et al. [7], adjusting kernel sizes can enhance spatial correlation extraction, particularly not beneficial for CSI matrices characterized by inherent randomness due to wireless channel conditions. Furthermore, Lu et al.'s introduction [7] of network binarization represents a novel approach, marking the first application of binary neural networks [8] in this domain.

Many studies in this field prioritize optimizing the NMSE, overlooking the practical implications of models in real-world scenarios involving imperfect channels in wireless communication. Factors such as short-term channel correlation and resource constraints are frequently overlooked, despite their considerable dependency on model runtime. The runtime of most models is heavily reliant on the Graphics Processing Unit (GPU) of the hardware on which these models are evaluated, and this runtime is proportional to the FLOPs. Most proposed models exhibit higher NMSE when considering a snapshot of CSI and assuming zero mobility of User Equipment (UE). To address this gap, we evaluate several state-of-the-art proposed models using the QuadriGa dataset [9]. Since most models are trained on the COST2100 [10] channel model, it's essential to consider its implications for generalization to real-world scenarios. We generate CSI samples while considering the mobility of UE, and we examine snapshots at various time intervals to assess the generalization capabilities of the proposed models. We quantified the number of FLOPs and parameters required for User Equipment (UE) in Table I. Notably, most proposed models demand at least one million parameters, emphasizing the need to consider resource limitations and computational efficiency alongside NMSE optimization in model design. One of our proposed approaches to mitigate the high parameter count is to utilize alternative compression techniques, acknowledging that it may result in a slight loss of accuracy. If a significant decrease in accuracy occurs, it will demonstrate the dependence of these models on the fully connected network.

TABLE I: Methods Parameters Comparison

Method	**UE Params**	**UE FLOPs**
CsiNet	1052K	1094K
CRNet	1049K	1235K
ACRNet-1x	1049K	1235K
CsiNetPlus	1049K	1462K
TransNet	2381K	1054K

References

[1] C. Wen et al., "Deep Learning for Massive MIMO CSI Feedback," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 748–751, 2018.

[2] Z. Lu et al., "Binarized Aggregated Network With Quantization," *IEEE Trans. Wireless Commun.*, vol. 21, no. 7, pp. 5514–5525, 2022.

[3] Z. Lu et al., "Multi-resolution CSI Feedback With Deep Learning," *ICC*, pp. 1–6, 2020.

[4] X. Yu et al., "DS-NLCsiNet: Exploiting Non-Local Neural Networks," *IEEE Commun. Lett.*, vol. 24, no. 12, pp. 2790–2794, 2020.

[5] A. Chakma et al., "Deep Decoder CsiNet for FDD Massive MIMO," *IEEE Wireless Commun. Lett.*, vol. 12, no. 12, pp. 2073–2077, 2023.

[6] Y. Cui et al., "TransNet: Full Attention Network for CSI Feedback," *IEEE Wireless Commun. Lett.*, vol. 11, no. 5, pp. 903–907, 2022.

[7] Z. Lu et al., "Binary Neural Network Aided CSI Feedback," *IEEE Wireless Commun. Lett.*, 2020.

[8] M. Nagel et al., "A White Paper on Neural Network Quantization," *arXiv:2106.08295*, 2021.

[9] F. Burkhardt et al., "QuaDRiGa: A MIMO Channel Model," *EuCAP*, pp. 1274–1278, 2014.

[10] L. Liu et al., "The COST 2100 MIMO Channel Model," *IEEE Wireless Commun.*, vol. 19, pp. 92–99, 2012.