Quantization Effects in a CNN-based Channel Estimator

Fábio D. L. Coutinho
Instituto de Telecomunicações, DETI
Universidade de Aveiro
Aveiro, Portugal
fabiocoutinho@ua.pt

Hugerles S. Silva
Instituto de Telecomunicações, DEE
Universidade de Brasília
Brasília, Brazil
hugerles.silva@av.it.pt

Petia Georgieva
Instituto de Telecomunicações, IEETA, DETI
Universidade de Aveiro
Aveiro, Portugal
petia@ua.pt

Arnaldo S. R. Oliveira
Instituto de Telecomunicações, DETI
Universidade de Aveiro
Aveiro, Portugal
arnaldo.oliveira@ua.pt

***Abstract*—In this paper, we study the impact of the convolutional neural networks (CNN) quantization for the channel estimation. In the wireless network edge, with the adoption of deep learning (DL) algorithms, the limited computational resources bottleneck needs to be considered. Thus, a study using a field-programmable gate array (FPGA) platform is carried out, where the resource utilization and the timing requirements are analyzed. A single-input single-output orthogonal frequency-division multiplexing (OFDM) end-to-end link is adopted in this work. The bit error rate (BER) measures the quantization impact of the CNN-based channel estimation on the global system. The obtained results show that an improvement in the maximum operating frequency and in the resource efficiency can be obtained without deteriorating the end-to-end performance.**

***Index Terms*—Channel Estimation; CNN; FPGA; OFDM; Quantization; Real-time systems.**

I. Introduction

Recently, novel deep learning (DL) algorithms have been developed for various processing functions of wireless communications systems [1]. One of those functions is the channel estimation, which presents several challenges due to nonlinear and complex issues between the transmitter and the receiver. For such purposes, a DL architecture particularly successful for the channel estimation is the convolutional neural network (CNN) [2]. The CNN's ability to automatically extract the underlying representative characteristics and features in image processing tasks is explored to improve the channel estimation [3]. However, the CNN hardware deployment is typically performed in a software-oriented approach, which imposes resource overhead in the edge of the radio access networks (RAN) due to the computation intensity of the floating-point architectures.

Therefore, an efficient solution to exploit the computational efficiency in the custom hardware is the bit quantization, which reduces the complexity and storage demands. As evidenced in [4], the prototyping step, which includes the quantization process, assumes an important process in the DL algorithms implementation in resource-limited hardware platforms, such the field-programmable gate array (FPGA) that is commonly used in processing functions on the RAN edge. Then, if the CNN architectures are optimized for the hardware implementation by means of the quantization, the CNN-based channel estimation algorithms can be implemented in real-time scenarios on the network edge.

In the literature, the quantization of CNN structures was addressed in [5], [6], in which low-precision weights and bias were exploited in order to decrease the hardware and memory requirements. Furthermore, an FPGA implementation in these papers was obtained to analyze the resource demands. In terms of edge processing functions, a study for modulation recognition was performed in [7], assessing the quantization of a CNN architecture to obtain an optimized FPGA implementation. Regarding the channel estimation, to the best of the author's knowledge, this is the first work in which the quantization effects of a CNN structure are analyzed from the design to the real-time hardware implementation. Hence, this work presents a study concerning the CNN quantization for the channel estimation with the real-time FPGA implementation as target. First, a CNN architecture is designed, followed by its flexible quantization. Several bit configurations are adopted to analyze the fixed-point quantization effects on the channel estimation performance. Furthermore, the resource utilization, the timing analysis and its system performance are obtained for each fixed-point FPGA implementation.

The remaining of the paper is organized as follows. In Section II, the considered system model is described. Section III provides a overview of the adopted CNN architecture in this work. Results are provided in Section IV. Section V is dedicated to the conclusions and future work.

This work is funded by FCT/MCTES through national funds and when applicable co-funded by EU funds under the project UIDB/50008/2020-UIDP/50008/2020, by the Project Augmented Humanity [POCI-01-0247-FEDER-046103 and LISBOA-01-0247-FEDER-046103], financed by Portugal 2020, under the Competitiveness and Internationalization Operational Program, the Lisbon Regional Operational Program, and the European Regional Development Fund, and by the FCT - Fundação para a Ciência e a Tecnologia, I.P. under the PhD scholarship ref. SFRH/BD/12897/2022.

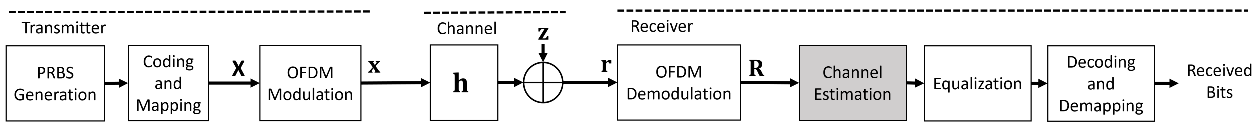

Fig. 1. The SISO OFDM system model adopted in this work.

Notation: We use boldface small letters and capital letters to denote vectors and matrices, respectively. The operator $\circledast$ represents circular convolution.

II. System Model

The system model considered in this work is a single-input single-output (SISO) orthogonal frequency division multiplexing (OFDM) end-to-end link under the tapped delay line (TDL) channel model, typically considered in the current generation of wireless networks. Regarding the channel, issues like the Doppler effect, fading and multipath impose the loss of subcarriers orthogonality in OFDM systems which deteriorates the global system performance. Then, the main goal of the channel estimation is to estimate the overall complex gain from the transmitter to the receiver.

In Fig. 1, an overview of the system model is depicted. After the pseudorandom binary sequence (PRBS) generation, the M-ary quadrature amplitude modulation (M-QAM) is performed, and then, the mapping of the frequency resource grid is realized ($\mathbf{X}$). The OFDM modulation is accomplished to acquire the time-domain signal ($\mathbf{x}$) from the resource grid. After the channel, the received signal ($\mathbf{r}$) in the time-domain can be expressed as $\mathbf{r} = \mathbf{x} \circledast \mathbf{h} + \mathbf{z}$, in which $\mathbf{h}$ is the channel impulse response, and $\mathbf{z}$ denotes the additive white Gaussian noise. In the receiver, we assume that the time and frequency synchronizations are perfect in order to individualize the channel estimation. After the OFDM demodulation, the received frequency resource grid ($\mathbf{R}$) is obtained as $\mathbf{R} = \mathbf{XH} + \mathbf{Z}$, in which the $\mathbf{X}$, $\mathbf{H}$ and $\mathbf{Z}$ is the frequency response of the $\mathbf{x}$, $\mathbf{h}$ and $\mathbf{z}$, respectively. It is worth mentioning that in this work, a pilot aided scheme is considered to decrease the overhead imposed by the time training symbol. After the channel estimation, the complex channel gain ($\hat{\mathbf{H}}$) is estimated, and it is used in the equalization to mitigate the channel effects. Finally, the decoding and demapping is performed to recover the transmitted bits and compute the bit error rate (BER).

III. CNN Architecture

The goal of this paper is to study the impact of the CNN quantization in the channel estimation. The CNN architecture we proposed for this processing function is inspired by [2], and is shown in Fig. 2. The network consists of two convolutional layers (Conv), filters with dimension 5×5 and ReLu activation functions. The CNN was trained offline using the machine learning toolbox of Matlab, to optimize the network parameters $\mathbf{\Theta}$. Dataset of 50000 (complex-values) examples was generated for the training process, with the real and imaginary parts computed separately.

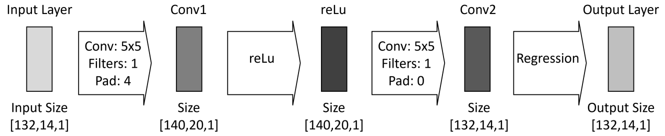

Fig. 2. The considered CNN architecture with two convolutional layers, and a reLu activation function used for channel estimation.

After obtaining the optimized $\mathbf{\Theta}$ parameters, a hardware-oriented implementation of the CNN architecture was performed. Therefore, an initial floating-point CNN model was designed in Matlab (Matlab) and from this, one floating-model ($\mathrm{Simulink}$) and five fixed-point real-time models in Simulink were implemented. In order to obtain the hardware description language (HDL) code for FPGA implementation, the HDL Coder is used due to its versatility to convert the fixed-point Simulink models. Furthermore, this tool uses the Vivado libraries to synthetize and implement the target design in a specific hardware platform.

The fixed-point configurations adopted in this work are 32 bits with 28 fractional places (Fix32_28), 24 bits with 20 fractional places (Fix24_20), 16 bits with 12 fractional place (Fix16_12), 12 bits with 8 fractional places (Fix12_8), and 8 bits with 4 fractional places (Fix8_4). Table I shows the mean-square error (MSE) of these implementations regarding the floating-point Matlab model as a reference. As we can see, the MSE increasing follows the decrease of bits length, as expected. Thus, in the next section, the effect of each quantizated model concerning the channel estimation is analyzed.

TABLE I
MSE between the different models.

Model	MSE	Model	MSE
Simulink	2.3×10^{-18}	Fix16_12	2.4×10^{-7}
Fix32_28	5.6×10^{-17}	Fix12_8	1.5×10^{-5}
Fix24_20	9.1×10^{-13}	Fix8_4	3.5×10^{-2}

IV. Results

In this section, the results of the FPGA implementation of each model is presented, in which the resource utilization and the timing requirements are analyzed. Next, the system impact of the data type quantization in the channel estimation is obtained. The adopted device is a xczu19eg-ffvb1517-1-e.

A. Resource Utilization and Timing Requirements

Concerning the resource usage study, Table II shows the implemented lookup table (LUT), LUT random access memory (LUTRAM), flip-flops (FF) and digital signal processing (DSP) components. As we can see, the used resources are highly dependent on the bit length of the model, which means that the resource overhead can be diminished if the correct quantization is performed.

TABLE II
RESOURCE ALLOCATED IN THE TARGET FPGA.

Resources	**LUT**	**LUTRAM**	**FF**	**DSP**
Available	522720	161280	1045440	1968
Fix32_28	499354	376	393806	200
	95.5%	**0.2%**	**37.7%**	**10.2%**
Fix24_20	379087	312	293881	100
	72.5%	**0.2%**	**28.1%**	**5.1%**
Fix16_12	251489	222	193996	50
	48.1%	**0.1%**	**18.6%**	**2.5%**
Fix12_8	186963	0	143913	50
	35.8%	**0%**	**13.8%**	**2.5%**
Fix8_4	125537	2	295243	0
	24%	**0.01%**	**9.1%**	**0%**

The maximum operation frequency is also influenced by the data size, as depicted in Table III. In addition, the period (T) by the critical data path of the CNN is provided and the corresponding maximum operation frequency (f). Although there is a certain tendency in the decrease of the data path time along the decrease of the data size, in the Fix12_8 model an increase is verified in the total path time. This can be the result of a different strategy of the implementation tool, which is correlated with the fact that the LUTRAMs are not used in this case.

TABLE III
TIMING REQUIREMENTS IN THE TARGET FPGA.

	Fix32_28	Fix24_20	Fix16_12	Fix12_8	Fix8_4
T (ns)	17.8	9.6	6.9	8.2	6.8
f (MHz)	56.1	103.8	144.5	122.4	147.5

B. Channel Estimation Impact in the End-to-end System

In addition to the trade-off study between the resource overhead and timing requirements, the impact of the CNN architecture quantization for the channel estimation needs to be analyzed in the end-to-end link performance by means of the BER. For this, a test dataset composed by 10000 examples is used. The impact of the quantization in the BER metric is shown in Fig. 3, in which the theoretical (Theo.), the traditional least-square (LS) and the least minimum mean-square error (LMMSE) estimators are also considered for comparison. Note that the LS algorithm is usually adopted in the real systems due to its simplicity, and the LMMSE is the optimum one but its practical implementation is prohibitive considering the need of channel knowledge.

The obtained results shows an immediately conclusion, the Fix8_4 architecture cannot perform the channel estimation. Thus, the adopted CNN configurations, except the Fix8_4, are prepared to implement the channel estimation. As we can see in Fig. 3, the BER results of these networks outperforms the LS estimator and approaches the optimum LMMSE, validatinng its use. With this study, a initial choice of a fixed-point CNN architecture for the channel estimation can be performed based on resources, timing and system performance results. Furthermore, the obtained BER curves are in accordance with a data input signal quantization study performed with a floating-point 32-bit software-oriented CNN in [8, Fig. 10].

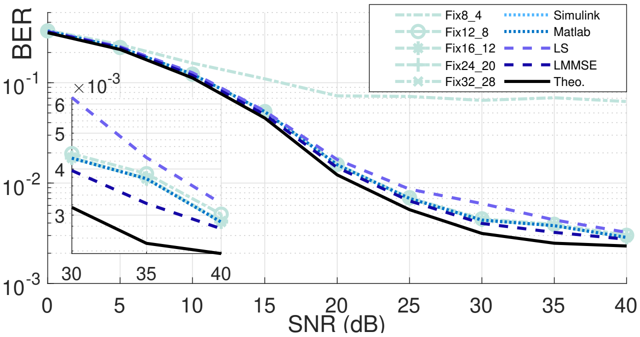

Fig. 3. BER as a function of signal noise ratio (SNR) regarding the channel estimation, considering the theoretical value, the LMMSE and LS estimators, and the quantizated CNN models. The modulation order adopted is $M = 16$.

V. CONCLUSIONS

In this work, a study regarding the convolutional neural networks (CNNs) quantization in the channel estimation was performed for the next generation of wireless communications, in which an optimized CNN architecture needs to be found for the network edge implementation. Five fixed-point configurations were analyzed concerning resource and timing requirements in a target hardware platform. Furthermore, the global end-to-end impact of the channel estimation performance was also studied in terms of bit error rate (BER). As future work, robust deep learning (DL) algorithms for channel estimation can be studied using novel DL-oriented platforms, as adaptive compute acceleration platforms.

REFERENCES

[1] W. Jiang, B. Han, M. A. Habibi and H. D. Schotten, "The road towards 6G: A comprehensive survey," *IEEE Open J. of the Commun. Soc.*, vol. 2, pp. 334-366, 2021.

[2] M. Soltani, V. Pourahmadi, A. Mirzaei and H. Sheikhzadeh, "Deep learning-based channel estimation," *IEEE Commun. Lett.*, vol. 23, no. 4, pp. 652-655, Apr. 2019.

[3] K. Mei, J. Liu, X. Zhang, K. Cao, N. Rajatheva and J. Wei, "A low complexity learning-based channel estimation for OFDM systems with online training," *IEEE Trans. Commun.*, vol. 69, no. 10, pp. 6722-6733, Oct. 2021.

[4] S. Ali, W. Saad, and D. Steinbach, "White paper on machine learning in 6G wireless communication networks," in *6G Research Visions*, University of Oulu, vol. 7, Apr. 2020.

[5] Y. Li, S. Lu, J. Luo, W. Pang and H. Liu, "High-performance convolutional neural network accelerator based on systolic arrays and quantization," in *Proc. of 2019 IEEE 4th International Conference on Signal and Image Processing (ICSIP)*, pp. 335-339, Oct. 2019.

[6] Y. -H. Wu, H. Lee, Y. S. Lin and S. -Y. Chien, "Accelerator design for vector quantized convolutional neural network," in *Proc. of 2019 IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, pp. 46-50, Jul. 2019.

[7] S. Kumar, R. Mahapatra and A. Singh, "Automatic modulation recognition: An FPGA implementation," *IEEE Commun. Lett.*, vol. 26, no. 9, pp. 2062-2066, Sept. 2022.

[8] F. D. L. Coutinho, H. S. Silva, P. Georgieva and A. S. R. Oliveira, "5G cascaded channel estimation using convolutional neural networks", *Digit. Signal Process.*, vol. 129, no. 103483, Jun. 2022.