Channel Estimation using Temporal Convolutional Networks for V2X Communications

Juan D. Jovane and Chia-Han Lee
Institute of Communications Engineering & Department of Electronics and Electrical Engineering
National Yang Ming Chiao Tung University, Taiwan

***Abstract*—To achieve satisfactory performance in vehicle-to-everything (V2X) communications, it is paramount to accurately estimate the channel. The traditional data-pilot aided (DPA) scheme and the variation of DPA, e.g., spectral temporal averaging (STA), have been adopted for IEEE 802.11p due to their low complexity, but their performances are not satisfactory. The more recently proposed time domain reliable test frequency domain interpolation (TRFI) scheme only marginally improves the performance. Deep neural network (DNN)-based estimators, e.g., STA-DNN and TRFI-DNN, have substantially improved the channel estimation, and the long short-term memory (LSTM)-based estimators, such as LSTM-DPA-TA and LSTM-MLP-DPA, achieve the state-of-the-art performance. LSTM-based estimators, however, have high computational complexity. In this paper, we propose a novel channel estimator that leverages temporal convolutional networks (TCNs) combined with the DPA procedure to estimate and track channel variations. Simulations on realistic V2X scenarios show that the proposed TCN-DPA channel estimation scheme outperforms existing methods in almost all V2X scenarios. The proposed estimator has about one order of magnitude improvement in terms of bit error rate compared to LSTM-based estimators. By exploiting the parallelism inherent in the TCN architecture, the computational complexity of the proposed TCN-DPA estimator is 40% and 47% lower than LSTM-DPA-TA and LSTM-MLP-DPA, respectively. Moreover, the training time of TCN-DPA is only 52% and 42% of the time of LSTM-DPA-TA and LSTM-MLP-DPA, respectively.**

I. Introduction

Vehicle-to-everything (V2X) communication has become critical for a wide variety of emergent applications. For example, developing reliable vehicle-to-vehicle (V2V) and roadside-to-vehicle (RVC) networks is essential for the future of autonomous vehicles [1]. Channel estimation and tracking for V2X communications is challenging due to user mobility and the time-varying nature of environments [2]. The orthogonal frequency-division multiplexing (OFDM) frame structure used in IEEE 802.11p [3] lacks sufficient pilots to provide reliable channel estimation—an initial accurate estimate is possible using the preamble at the beginning of the frame, but it becomes outdated as frames get longer. Therefore, estimators based on the data-pilot aided (DPA) estimation, where each subcarrier demapped utilizing the channel estimate from the previous OFDM symbol is used for updating the channel estimate for the current OFDM symbol, is conventionally adopted in IEEE 802.11p. Several improvements over the DPA scheme have been proposed, but they all have their own drawbacks. For instance, the spectral temporal averaging (STA) scheme [4], which introduces a time-averaging (TA) filter, performs well at low signal-to-noise ratio (SNR) regimes. However, STA tends to underperform in high SNR regions where the filter may generate an averaging error. The constructed data pilots (CDP) scheme [5] leverages the correlation between the adjacent OFDM symbols to establish a criterion to determine whether a DPA estimate is reliable and needs to be updated. CDP achieves a better performance than STA, but usually performs poorly when higher-order modulations are adopted. The time domain reliable test frequency domain interpolation (TRFI) scheme [6] determines whether the DPA estimates are reliable by the cubic-spline interpolation using reliable subcarriers, but its performance is still unsatisfactory.

Recently, deep learning (DL) has become popular for solving physical layer problems in communications. By the introduction of deep neural networks (DNNs) for post-processing after the DPA estimation, such as the schemes STA-DNN [7] and TRFI-DNN [8], significant improvements are observed. The estimators employing long short-term memory (LSTM) architectures alongside DPA, such as LSTM-MLP-DPA [9] and LSTM-DPA-TA [10], are the state-of-the-art in terms of bit error rate (BER) performance. Nevertheless, LSTM-based methods suffer from the significant computational burden since each time-step has to be processed serially during training.

Temporal convolutional networks (TCNs) have recently been proven to be able to compete with notable sequence predicting networks such as LSTMs and gated recurrent units (GRUs) for applications including segmentation, language processing, and translation tasks [11]. In this paper, we propose the TCN-DPA channel estimation scheme based on the combination of the TCN network and the DPA procedure. The TCN is in charge of sequence prediction and noise removal at the frequency axis while the DPA procedure aids the network by correlating subsequent OFDM symbols. The proposed TCN-DPA channel estimation scheme has been evaluated on realistic V2X scenarios. Simulation results show that the proposed TCN-DPA channel estimator outperforms existing methods in almost all V2X scenarios. The proposed TCN-DPA has one order of magnitude improvement in terms of BER compared to either LSTM-DPA-TA or LSTM-MLP-DPA. The proposed TCN-DPA method also tackles the computational complexity issue in LSTM-based estimators by exploiting the parallelism inherent in the TCN architecture. The complexity of the proposed TCN-DPA estimator is 40% and 47% lower than LSTM-DPA-TA and LSTM-MLP-DPA, respectively. Moreover, TCN-DPA considerably reduces the training time of the network

down to 52% and 42% of the time required by LSTM-DPA-TA and LSTM-MLP-DPA, respectively.

II. System Model

The OFDM frame structure used for V2X communications is specified in IEEE 802.11p [3]. The frame has 64 subcarriers, indexed by $\mathcal{S} = \{1, 2, \ldots, 64\}$. Each OFDM symbol is composed of 12 null subcarriers indexed by $\mathcal{S}_E = \{1, \ldots, 6, 33, 60, \ldots, 64\}$, 4 pilot subcarriers indexed by $\mathcal{S}_P = \{12, 26, 40, 54\}$, and 48 data subcarriers index by $\mathcal{S}_D$. The preamble is formed by two known OFDM symbols at the beginning of the frame. Perfect synchronization is assumed. The received signal at the frequency domain can then be written as

$$\boldsymbol{y}_i[k] = \boldsymbol{h}_i[k]\boldsymbol{x}_i[k] + \boldsymbol{n}_i[k], \tag{1}$$

where k and i respectively represent the k-th subcarrier at the i-th time instant, $\boldsymbol{h}_i[k]$ is the channel frequency response, $\boldsymbol{x}_i[k]$ is the transmitted constellation symbol, and $\boldsymbol{n}_i[k] \sim \mathcal{CN}(0, \sigma^2)$ is additive complex white Gaussian noise with variance σ^2.

III. Review of Existing Channel Estimators

Popular channel estimation schemes for vehicular communications include the following.

- ***DPA***: The first step is to obtain an initial estimate for the frame, denoted by $\hat{\boldsymbol{h}}_0$. This can be attained by least square (LS) estimation at the preamble, i.e.,

$$\hat{\boldsymbol{h}}_0[k] = \frac{\boldsymbol{y}_1[k] + \boldsymbol{y}_2[k]}{2\boldsymbol{p}[k]}, \tag{2}$$

  where $\boldsymbol{p}$ represents the known preamble, and $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ are the first two received symbols the preamble is located. For the remaining OFDM symbols, the equalized symbol, $\boldsymbol{y}_i^{\text{eq}}$, can be estimated as

$$\boldsymbol{y}_i^{\text{eq}}[k] = \frac{\boldsymbol{y}_i[k]}{\hat{\boldsymbol{h}}_{i-1}^{\text{DPA}}[k]}, \quad \hat{\boldsymbol{h}}_0^{\text{DPA}}[k] = \hat{\boldsymbol{h}}_0[k]. \tag{3}$$

  Afterwards, the equalized received symbol is mapped to the its closest point in the constellation diagram and obtain $\boldsymbol{q}_i[k] = \text{map}(\boldsymbol{y}_i^{\text{eq}}[k])$, where $\text{map}(\cdot)$ is the operation for finding the closest constellation point. Finally, the channel update is given by

$$\hat{\boldsymbol{h}}_i^{\text{DPA}}[k] = \frac{\boldsymbol{y}_i[k]}{\boldsymbol{q}_i[k]}. \tag{4}$$

- ***STA***: All steps of the DPA procedure are executed. However, the frequency domain averaging is performed as

$$\hat{\boldsymbol{h}}_i^{\text{FD}}[k] = \sum_{\lambda=-\beta}^{\beta} \bar{\omega}\hat{\boldsymbol{h}}_i^{\text{DPA}}[k+\lambda], \tag{5}$$

  where $\bar{\omega} = \frac{1}{2\beta+1}$ and $2\beta + 1$ represents the amount of subcarrier symbols to be averaged. Finally, an auto-regressive time filter is applied to obtain the channel update as

$$\hat{\boldsymbol{h}}_i^{\text{STA}}[k] = \left(1 - \frac{1}{\alpha}\right)\hat{\boldsymbol{h}}_{i-1}^{\text{STA}}[k] + \frac{1}{\alpha}\hat{\boldsymbol{h}}_i^{\text{FD}}[k], \tag{6}$$

  where α is a filter parameter.
- ***CDP***: The current DPA and the previous CDP estimates are used to equalize the previously received symbol as

$$\bar{\boldsymbol{y}}_{i-1}^{\text{eq}}[k] = \frac{\boldsymbol{y}_{i-1}[k]}{\hat{\boldsymbol{h}}_i^{\text{DPA}}[k]}, \quad \tilde{\boldsymbol{y}}_{i-1}^{\text{eq}}[k] = \frac{\boldsymbol{y}_{i-1}[k]}{\hat{\boldsymbol{h}}_{i-1}^{\text{CDP}}[k]}, \tag{7}$$

  where $\hat{\boldsymbol{h}}_i^{\text{CDP}}[k]$ represents the CDP channel estimate. The equalized symbols, $\bar{\boldsymbol{y}}_{i-1}^{\text{eq}}[k]$ and $\tilde{\boldsymbol{y}}_{i-1}^{\text{eq}}[k]$, are then demapped to their closest constellation point as $\bar{\boldsymbol{q}}_{i-1}[k] = \text{map}(\bar{\boldsymbol{y}}_{i-1}^{\text{eq}}[k])$ and $\tilde{\boldsymbol{q}}_{i-1}[k] = \text{map}(\tilde{\boldsymbol{y}}_{i-1}^{\text{eq}}[k])$, respectively. Finally, the channel is updated according to the criteria

$$\hat{\boldsymbol{h}}_i^{\text{CDP}}[k] = \begin{cases} \hat{\boldsymbol{h}}_{i-1}^{\text{CDP}}[k], & \text{if } \bar{\boldsymbol{q}}_{i-1}[k] \neq \tilde{\boldsymbol{q}}_{i-1}[k], \\ \hat{\boldsymbol{h}}_i^{\text{DPA}}[k], & \text{otherwise.} \end{cases} \tag{8}$$

- ***TRFI***: The steps of this process are very similar to the ones performed by the CDP estimator, but the criteria (8) is replaced by the cubic spline interpolation to generate updates for the unreliable subcarriers.
- ***DNN-based estimators***: For STA-DNN and TRFI-DNN, fully-connected layers are added to improve the performance of STA and TRFI estimators.
- ***LSTM-based estimators***: LSTM-based estimators, including LSTM-MLP-DPA and LSTM-DPA-TA, are state-of-the-art estimators for vehicular communications. In LSTM-MLP-DPA, the received OFDM symbol is input to an LSTM unit, and then the output is filtered by a maximum likelihood perceptron (MLP). Finally, the estimate from the neural network is helped by the classical DPA for channel tracking. In LSTM-DPA-TA, the received OFDM symbol is input to an LSTM unit, and then the DPA procedure is executed. Finally, a time-averaging filter is applied.

IV. Proposed Channel Estimator

In this section, we describe the TCN and the design of the proposed TCN-DPA estimator. Complexity of various estimators is then analyzed.

A. Temporal Convolutional Network (TCN)

A TCN is a convolutional neural network (CNN)-based alternative to recurrent neural networks (RNNs) that can capture the features required for sequence prediction [11]. In a TCN block, all operations must be causal. To meet these criteria, a padding with length (kernel size $-$ 1) is inserted at the beginning of the sequence. This padding ensures that the output of every hidden layer in the TCN block has the same size of the input. Furthermore, it forces the prediction of the element y_i in the sequence $\mathbf{Y}$ of length T' to be only

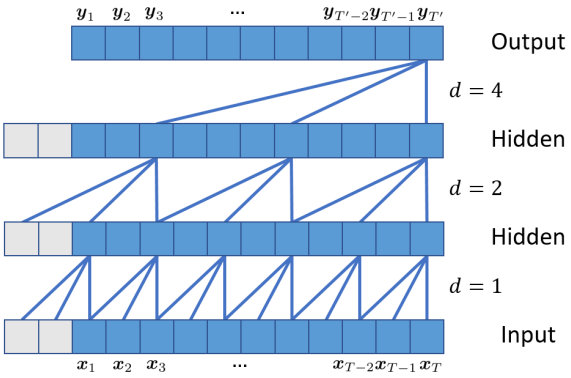


Fig. 1: An example of causal dilated convolution.

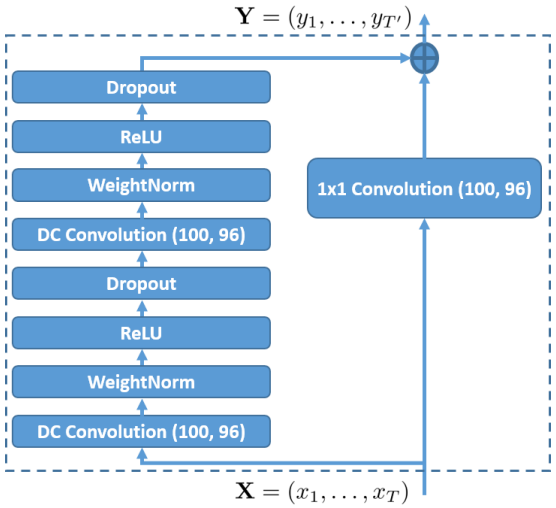


Fig. 2: TCN residual block (adapted from [11]).

dependent on elements $x_1, \ldots, x_i$, $i \leq T$, from the sequence $\mathbf{X}$ of length T.

In order to have more control over the effective history (i.e., the extent in which a neural network is able to look into the past), convolutions used in TCNs are dilated, controlling the time delay in-between subsequent elements of the kernel. A dilated convolution with dilation d, denoted as $*_d$, for the element x_i and a kernel w of size W can be written as $y_i = (x *_d w)(i) = \sum_{j=0}^{W-1} w(j) x_{i-d \cdot j}$. An example of dilated convolution is shown in Fig. 1.

The final element which completes a TCN residual block (RB) is the addition of residual connections. An RB always contains at least two branches: one is the original input $\mathbf{X}$ and the other is the output $\mathbf{Y}$ after going through a set of transformations $\mathcal{F}$. The final output of the RB with activation z is given by $\mathbf{Y} = z(\mathbf{X} + \mathcal{F}(\mathbf{X}))$. The TCN residual block is shown in Fig. 2.

B. Proposed TCN-DPA

The proposed TCN-DPA channel estimator combines the TCN architecture and the DPA procedure. In TCN-DPA, the neural network predicts a frequency axis sequence based on another frequency axis sequence. The temporal correlation

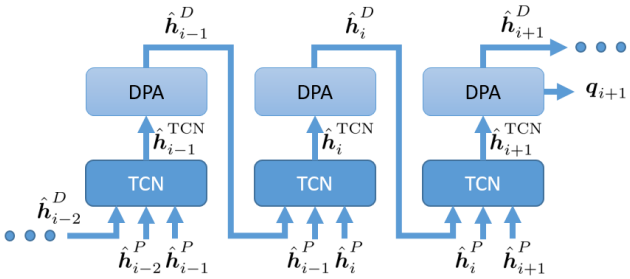


Fig. 3: Proposed TCN-DPA estimator.

is then performed by the DPA procedure. The TCN-DPA estimation can be obtained after executing the following steps:

1) First, the least square estimate with the first two symbols in the frame is performed to obtain an initial estimate $\hat{\boldsymbol{h}}_0$ as in (2). This will serve as the initial input of the network.
2) After the initial estimation is finished, the compensation is done by a set of TCN residual blocks. Note that subcarriers in LSTM-based estimators are considered features of a time-step in a sequence. However, in the proposed TCN-DPA, each subcarrier is considered a time-step. Denote $\hat{\boldsymbol{h}}_i^D$ as the vector of estimated channels for data subcarriers which is constructed by collecting $\hat{\boldsymbol{h}}_i[k]$, $k \in \mathcal{S}_D$. Similarly, denote $\hat{\boldsymbol{h}}_i^P$ as the vector of estimated channel for pilot subcarriers which is constructed by collecting $\hat{\boldsymbol{h}}_i[k]$, $k \in \mathcal{S}_P$. Then the output of TCN $\hat{\boldsymbol{h}}_i^{\text{TCN}} = f_{\text{TCN}}(\theta; \hat{\boldsymbol{h}}_{i-1}^D, \hat{\boldsymbol{h}}_{i-1}^P, \hat{\boldsymbol{h}}_i^P)$, where f_{TCN} denotes the function of TCN and θ are the weights. Note that the length of $\hat{\boldsymbol{h}}_i^{\text{TCN}}$ is the number of data subcarriers, N_D.
3) Afterwards, the DPA procedure utilizes $\hat{\boldsymbol{h}}_i^{\text{TCN}}$ to equalize and determine the received signal as

$$\boldsymbol{q}_i[k] = \text{map}\left(\frac{\boldsymbol{y}_i[k]}{\hat{\boldsymbol{h}}_i^{\text{TCN}}[k']}\right), \tag{9}$$

where $k \in \mathcal{S}_D$ and k' indexes the k'-th data subcarrier.
4) Finally, the estimation at the i-th time instant is updated using (4).

The proposed TCN-DPA is shown in Fig. 3. Note that the parallelism in the architecture leads to the reduced training time in comparison to LSTM-based estimators that require to input each OFDM symbol in a serial fashion during training.

C. Computational Complexity

The complexity of estimators is analyzed in terms to real-valued operations. Since the performance of traditional estimators is inferior to DL-based estimators, we focus on analyzing DL-based estimators. Firstly, we analyze the DPA estimation scheme. DPA first performs an LS estimate at the first two BPSK modulated symbols in the frame, which requires $2N_A$ real-valued summations and $2N_A$ real-valued divisions; N_A represents the total number of data and pilot subcarriers. The

TABLE I: Computational Complexity of Estimators

Estimator	Computational complexity
TRFI-DNN	$2\sum_{\ell=2}^{L} M_{\ell-1}M_{\ell} + 56N_U + 48N_A$
STA-DNN	$2\sum_{\ell=2}^{L} M_{\ell-1}M_{\ell} + 32N_A + 12N_D, M_1 = 2N_I$
LSTM-MLP -DPA	$4\left[L_{\text{LSTM}}^2 + L_{\text{LSTM}}(N_I+4) + N_I - 2\right]$ +2$\sum_{\ell=2}^{L} M_{\ell-1}M_{\ell} + 26N_A, M_1 = 2N_I$
LSTM-DPA-TA	$4\left[L_{\text{LSTM}}^2 + L_{\text{LSTM}}(N_I+4) + N_I - 2\right] + 32N_A$
TCN-DPA	$N_D^2(W+1) + N_D(2N_IW+3) + 26N_A$

remaining steps of DPA require two complex divisions, with each complex division needing $11N_A$ real-valued operations. Thus, DPA requires $26N_A$ operations in total.

The complexity of the TCN residual block depends mainly on the 1D convolutional layers. Each layer has a complexity determined by the number of input filters N_{FI}, the number of output filters N_{FO}, and the kernel size W. Each layer requires $N_{FI}N_{FO}W$ real-valued operations and N_{FO} real-valued summations. TCN's specific structure requires all hidden layer operations to have the same number of output filters, and with the number of output filters N_{FO} being the same as the number of data subcarriers N_D, the complexity of the TCN residual block is $[(2N_IW + N_D(W+1)+3]N_D = N_D^2(W+1) + N_D(2N_IW+3)$, where N_I is the number of subcarriers input to the network. Therefore, the total complexity of TCN-DPA is $N_D^2(W+1) + N_D(2N_IW+3) + 26N_A$.

Similar calculations for the complexity of other DL-based estimators can be done, but due to space limitation we only summarize the results in TABLE I. The terms L, M_ℓ, N_U, and L_{LSTM} respectively denote the number of hidden layers in the fully-connected network, the number of neurons in the ℓ-th fully-connected layer, the number of unreliable subcarriers specified by TRFI, and the number of hidden layers in LSTM. Note that N_I in different schemes may have different values, depending on the structure of the network.

V. Performance Evaluation

In this section, the BER performance of the proposed TCN-DPA is evaluated and compared with existing channel estimators. A complexity comparison against other DL-based estimators is also provided.

The vehicular channel models used for simulations are as follows [2]:

- **V2V-EX**: Two vehicles enter the expressway and move in opposite directions at 104 km/h. There is no obstruction between cars.
- **V2V-SDWW**: Two cars establish communication in an expressway with a middle wall separating lanes of the opposite directions. The vehicles were 300 m to 400 m away with the speed of 104 km/h.
- **V2V-UC**: Two vehicles move in opposite directions at 40 km/h due to traffic jam in the city.
- **R2V-EX**: The base station is located at the expressway with a vehicle moving at 104 km/h. The antenna is about 6.1 m high.

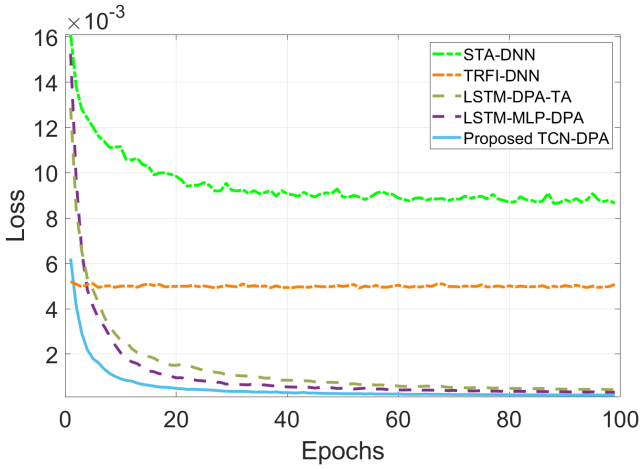


Fig. 4: Training loss of estimators for V2V-UC.

- **R2V-SS**: A base station is at an intersection in the suburban. The vehicle's velocity is about 40 km/h at a distance around 100 m. The antenna's height is around 6.1 m.
- **R2V-UC**: The base station is on the roadside in a urban canyon where a car moves at 40 km/h and the distance is about 100 m.

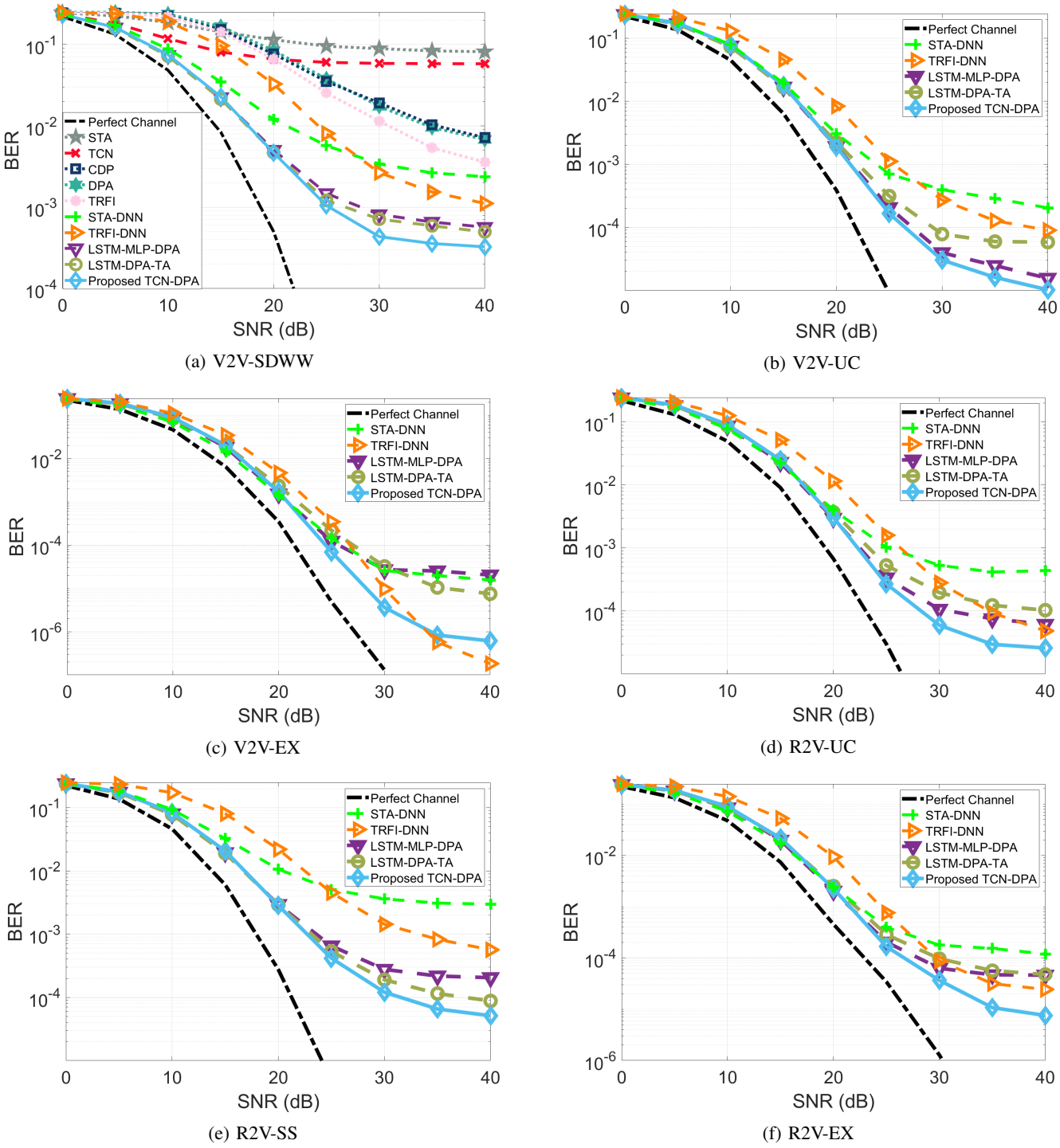
The OFDM frame is composed of 50 OFDM symbols with 64 subcarriers. 16-QAM modulation is used. The bandwidth is 10 MHz and the carrier frequency is 5.9 GHz.

The proposed TCN-DPA scheme has a 2-layer TCN residual block with layer dilations of $(1, 2)$ and a kernel of size 2. For the comparing estimators, the DNN setting used for STA-DNN and TRFI-DNN schemes follow the ones given in [7], where the neural network is composed of three hidden layers with 15 neurons each. The LSTM setting for LSTM-DPA-TA and LSTM-DPA-MLP schemes follow [10], where each LSTM has 128 hidden units. The MLP is composed of 2 layers with 128 and 40 neurons each. The number of testing and training frames is 4000 and 16000, respectively. The optimization of the neural networks is done using the ADAM optimizer with a learning rate of 0.001 and the batch size of 128 samples. The mean square error (MSE) serves as the loss function, and the training SNR is 40 dB.

In Fig. 4, we show the training of various estimators for the V2V-UC scenario as an example. As it can be observed, the proposed TCN-DPA converges faster than other schemes and has lower loss throughout the training.

A. BER Performance

Fig. 5 illustrates the outstanding performance of the proposed TCN-DPA estimator compared to existing estimators under various V2X channels. TCN-DPA constantly outperforms existing methods in almost all V2X scenarios. Firstly in Fig. 5(a), we present a BER comparison of all traditional and DL-based estimators for the V2V-SDWW scenario. DL-based estimators clearly have advantages over traditional estimators. For DL-based estimators, it is observable that DNN-based



(a) V2V-SDWW (b) V2V-UC

(c) V2V-EX (d) R2V-UC

(e) R2V-SS (f) R2V-EX

Fig. 5: BER of various estimators.

schemes can hardly compete with LSTM-based estimators. Our proposed TCN-DPA has similar performance with LSTM-based estimators at SNR smaller than 20 dB and outperforms them at SNR larger than 20 dB. At 40 dB SNR, the BER of the proposed TCN-DPA is 1.75 times lower than that of the second best estimator, LSTM-DPA-TA. We also compare TCN-DPA with a standalone TCN to prove the necessity of DPA. Since the performance of traditional schemes is inferior to DL-based schemes and for the clarity of figures, traditional schemes will not be compared in the remaining scenarios.

For the V2V-UC scenario shown in Fig. 5(b), LSTM-based estimators perform better than DNN-based estimators. In addition, LSTM-MLP-DPA is better than LSTM-DPA-TA. However, the proposed TCN-DPA is still superior than

TABLE II: Computational Complexity

Estimator	Real-valued operations
TRFI-DNN	12092
STA-DNN	11020
TCN-DPA (proposed)	73648
LSTM-DPA-TA	123944
LSTM-MLP-DPA	153528

LSTM-MLP-DPA, especially for SNR higher than 30 dB. For the V2V-EX scenario as shown in Fig. 5(c), the TRFI-DNN estimator performs unexpectedly well, outperforming all estimators at high SNR. However, for SNR smaller than 30 dB, the proposed TCN-DPA performs the best. TCN-DPA also performs significantly better than LSTM-based and DNN-based estimators for all SNR. As seen in Fig. 5(d) for the R2V-UC scenario, the proposed TCN-DPA again outperforms all existing estimators. Although the TRFI-DNN scheme outperforms LSTM-based schemes at high SNR, it is still inferior to the proposed TCN-DPA. For the R2V-SS scenario, Fig. 5(e) shows that while the LSTM-based estimators again outperforms the DNN-based estimators, the proposed TCN-DPA outperforms them all. Finally for the R2V-EX scenario, Fig. 5(f) illustrates the performance advantage of TCN-DPA. Although the TRFI-DNN scheme outperforms LSTM-based schemes at high SNR, the proposed TCN-DPA outperforms all existing estimators throughout all SNR.

B. Complexity Comparison

LSTM-MLP-DPA requires an LSTM unit that has $L_{\text{LSTM}} = 128$ hidden units and has an input of size $N_I = 112$, where the output sequence of such network is followed by a hidden layer with 40 neurons [9]. LSTM-DPA-TA has also an LSTM structure with $L_{\text{LSTM}} = 128$ hidden units, but in this case the input size is reduced to $N_I = 104$ since pilots from the last estimated symbol are not included when making an estimation [10]. For the proposed TCN-DPA, having only one residual block renders a shallow network with low complexity. The TCN-DPA scheme uses a kernel of size $W = 2$ and the number of complex data subcarriers $N_D = 96$, with a total input size of $N_I = 112$. For all cases, the number of active subcarriers per time frame is $N_A = 104$. As shown in TABLE II, the proposed TCN-DPA has lower complexity than both LSTM-MLP-DPA and LSTM-DPA-TA.

As compared in Fig. 6, the training time of LSTM-MLP-DPA and LSTM-DPA-TA is respectively 2.4 and 1.92 times of the training time of the proposed TCN-DPA. This shows the training advantage of TCN-DPA.

VI. Conclusion

In this paper, we have proposed a novel TCN-DPA estimator that combines TCN and the DPA procedure and delivers outstanding BER performance for V2X communications. Simulations on realistic V2X scenarios show that the proposed TCN-DPA outperforms existing methods in almost all V2X scenarios. The proposed estimator has at least one order of magnitude improvement in terms of BER compared to DNN-based and LSTM-based estimators. By exploiting the parallelism inherent in the TCN architecture, the proposed TCN-DPA has much lower complexity than LSTM-DPA-TA and LSTM-DPA-MLP. The proposed TCN-DPA also converges much faster than other DL-based estimators.

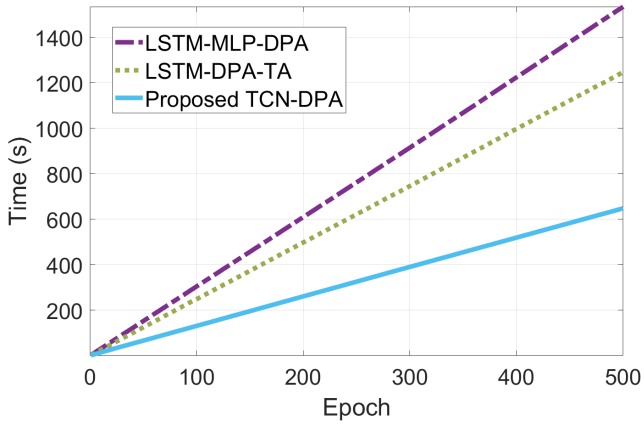


Fig. 6: Comparison of training time.

Acknowledgment

The support from National Science and Technology Council (NSTC), Taiwan under project 110-2221-E-A49-022-MY3 is gratefully appreciated.

References

[1] M. N. Ahangar, Q. Z. Ahmed, F. A. Khan, and M. Hafeez, "A survey of autonomous vehicles: Enabling communication technologies and challenges," *Sensors*, vol. 21, no. 3, 2021.

[2] G. Acosta-Marum and M. A. Ingram, "Six time and frequency-selective empirical channel models for vehicular wireless LANs," *IEEE Veh. Technol. Mag.*, vol. 2, no. 4, pp. 4–11, 2007.

[3] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an international standard for wireless access in vehicular environments," in *Proc. IEEE VTC Spring*, 2008, pp. 2036–2040.

[4] J. A. Fernandez, K. Borries, L. Cheng, B. V. Kumar, D. D. Stancil, and F. Bai, "Performance of the 802.11p physical layer in vehicle-to-vehicle environments," *IEEE Trans. Veh. Technol.*, vol. 61, no. 1, pp. 3–14, 2011.

[5] Z. Zhao, X. Cheng, M. Wen, B. Jiao, and C.-X. Wang, "Channel estimation schemes for IEEE 802.11p standard," *IEEE Intell. Transp. Syst. Mag.*, vol. 5, no. 4, pp. 38–49, 2013.

[6] Y.-K. Kim, J.-M. Oh, Y.-H. Shin, and C. Mun, "Time and frequency domain channel estimation scheme for IEEE 802.11p," in *Proc. 17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2014, pp. 1085–1090.

[7] A. K. Gizzini, M. Chafii, A. Nimr, and G. Fettweis, "Deep learning based channel estimation schemes for IEEE 802.11p standard," *IEEE Access*, vol. 8, pp. 113 751–113 765, 2020.

[8] ——, "Joint TRFI and deep learning for vehicular channel estimation," in *Proc. IEEE GLOBECOM Workshops (GC Wkshps)*, 2020, pp. 1–6.

[9] J. Pan, H. Shan, R. Li, Y. Wu, W. Wu, and T. Q. Quek, "Channel estimation based on deep learning in vehicle-to-everything environments," *IEEE Communications Letters*, vol. 25, no. 6, pp. 1891–1895, 2021.

[10] A. K. Gizzini, M. Chafii, S. Ehsanfar, and R. M. Shubair, "Temporal averaging LSTM-based channel estimation scheme for IEEE 802.11p standard," in *Prof. IEEE GLOBECOM*, 2021, pp. 1–7.

[11] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.