

High-Speed Channel Modeling with Deep Neural Network for Signal Integrity Analysis

Tianjian Lu, Ken Wu, and Zhiping Yang
 Google Inc.
 1600 Amphitheatre Parkway,
 Mountain View, California 94043
 Email: {tianjianlu, kenzwu, zhipingyang}@google.com

Ju Sun
 Stanford University
 450 Serra Mall, Building 380
 Stanford, CA 94305-2125
 Email: sunju@stanford.edu

Abstract—In this work, deep neural networks (DNNs) are trained and used to model high-speed channels for signal integrity analysis. The DNN models predict eye-diagram metrics by taking advantage of the large amount of simulation results made available in a previous design or at an earlier design stage. The proposed DNN models characterize high-speed channels through extrapolation with saved coefficients, which requires no complex simulations and can be achieved in a highly efficient manner. It is demonstrated through numerical examples that the proposed DNN models achieve good accuracy in predicting eye-diagram metrics from input design parameters. In the DNN models, no assumptions are made on the distributions of and the interactions among individual design parameters.

I. INTRODUCTION

A high-speed, chip-to-chip system-level design as shown in Fig. 1(a) requires intensive simulations for verification and optimization prior to manufacturing. The simulation techniques for signal integrity analysis include electromagnetic field solvers for component-level model extraction and circuit simulators for eye-diagram generation and system-level evaluation. During the manufacturing process, variations of design parameters often occur and alter the characteristics of a well-designed channel. Therefore, simulations are also heavily relied on in understanding the effects brought by manufacturing tolerances on signal integrity. Consequently, a large amount of data is made available through the aforementioned many iterations of simulations at various design stages.

The simulation techniques employed in characterizing high-speed channels for signal integrity can be very computationally expensive. There are efforts in utilizing domain decomposition schemes and parallel computing to enhance the efficiency of electromagnetic solvers on model extraction [1]. There are also approaches to efficiently generate eye diagrams by utilizing shorter data patterns as inputs [2]. The efficiency can also be improved if the simulation results obtained in a previous design or at an earlier design stage can be reused. In this work, we propose using deep neural networks (DNNs) to model high-speed channels for signal integrity analysis by taking advantage of the large amount of data obtained in the design process. DNN has recently made great progresses in selecting relevant results in web search, making recommendations in online shopping, identifying objects in images, and transcribing speech into text—to name a few [3], [4].

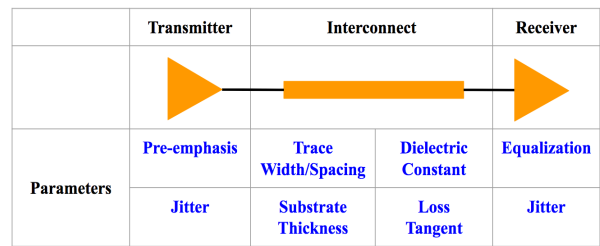
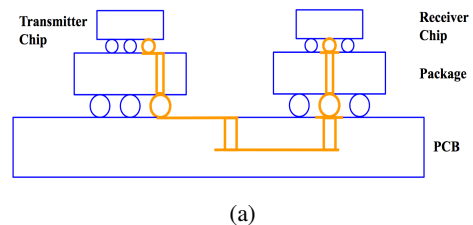


Fig. 1: (a) The topology of a high-speed channel and (b) the design parameters considered in this work.

To model high-speed channels, a DNN takes in raw design parameters as the input level and gradually transforms them into representations of higher and more abstract levels. With adequate number of such transformations, the DNN learns to predict eye-diagram metrics of a high-speed channel from its design parameters. A DNN model characterizes the high-speed channel through extrapolation with saved coefficients, which requires no complex simulations or substantial domain knowledge and can be achieved in a highly efficient manner. It is worth mentioning that in a DNN channel model, there are no assumptions made on the distributions of and the interactions among individual design parameters. The implementation of DNN in this work is based on Google’s TensorFlow [5].

II. DNN CHANNEL MODEL

Figure 1(a) shows the topology of a high-speed channel consisting of a transmitter, a receiver, and the interconnects in between. The eye height and width as illustrated in Fig. 2 are often used to assess signal integrity of a high-speed channel. Figure 1(b) tabulates the design parameters of a high-speed channel considered in this work. To design a high-speed chan-

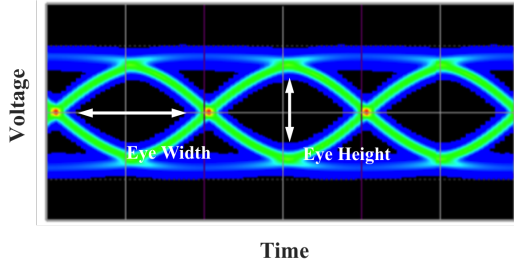


Fig. 2: Illustration of eye height and width in an eye diagram.

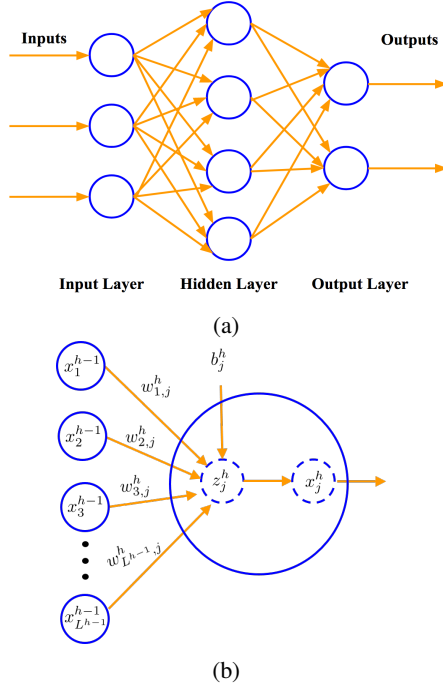


Fig. 3: (a) A feedforward neural network consists of one input layer, one hidden layer, and one output layer and (b) a node connects the $(h - 1)^{\text{th}}$ layer to the h^{th} layer.

nel, electromagnetic field solvers and eye-diagram generators are used to validate and optimize the design parameters and to address the corresponding manufacturing tolerances. All the simulation results obtained in the design process are saved as data sets to train and validate the DNN models.

As shown in Fig. 3(a), a feedforward neural network consists of many connected nodes in multiple layers. The number of nodes belonging to individual layers can be lumped into a vector $\{L\} = (L^{\text{in}}, L^1, \dots, L^h, \dots, L^n, L^{\text{out}})$ where L^{in} , L^h , L^{out} denote the number of nodes in the input layer, the h^{th} hidden layer, and the output layer, respectively. One system-level simulation produces one training or validation example $(\{x\}, \{\hat{y}\})$, where $\{x\}$ represents the input vector of design parameters and $\{\hat{y}\}$ is the target output vector. The input to the h^{th} hidden layer can be found through

$$\{z^h\} = \{x^{h-1}\} [W^h], \quad (1)$$

where the $L^{h-1} \times L^h$ matrix $[W^h]$ contains the weights

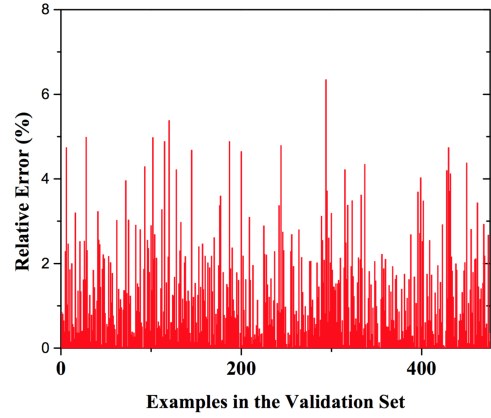


Fig. 4: Relative error of predicted eye height from the DNN model on the validation set.

mapping nodes on two adjacent hidden layers. It is worth mentioning that the bias term shown in Fig. 3(b) is ignored for simplicity. With the nonlinear activation function f_a , one obtains the output vector $\{x^h\}$ of the h^{th} hidden layer. Based on the input vector $\{x\}$, the described feedforward mechanism generates the output vector $\{y\}$, which at first is often very different from $\{\hat{y}\}$ with the relative difference defined by

$$\{e\} = \{y\} - \{\hat{y}\}. \quad (2)$$

To make $\{y\}$ a good approximation of $\{\hat{y}\}$, one needs to minimize the cost function

$$E = \frac{1}{2} \{e\} \{e\}^T. \quad (3)$$

The weights stored in matrix $[W]$ are the tunable parameters in minimizing the quadratic error E . To find a local minimum of the cost function, a backpropagation method is used. In the backpropagation method, it uses matrix $[D]$ to store the derivatives of the activation function and vector $\{\delta\}$ for the errors. The backpropagated error in the h^{th} hidden layer can be written as

$$\{\delta^h\} = \{\delta^{h+1}\} [W^{h+1}]^T [D^h]. \quad (4)$$

With the backpropagated error, the weights can be updated through

$$[W^h] = [W^h] - \gamma \{x^{h-1}\}^T \{\delta^h\}, \quad (5)$$

where γ is the learning rate.

III. NUMERICAL EXAMPLE

A deep neural network (DNN) is trained to predict the eye height. The DNN has three hidden layers of 100, 300, and 200 nodes, respectively. The learning rate is chosen as 0.01 and batch size is 25. The training and validation sets have 717 and 476 examples, respectively. The maximum number of iterations is set to 4000. The eye height in the data set varies from 148 to 253 mV. Table I shows the root-mean-square errors (RMSEs) and the maximum relative errors at the last iteration under three different optimizers, namely, the

TABLE I: Accuracy of predicted eye heights from the DNN model.

		Gradient Descent	Momentum	RMSProp
On Training Set	RMSE (mV)	3.1	1.9	2.6
	Maximum Relative Error (%)	6.2	4.1	5.2
On Validation Set	RMSE (mV)	3.4	2.7	3.1
	Maximum Relative Error (%)	5.9	6.1	6.3

TABLE II: Accuracy of predicted eye widths from the DNN model. A unit interval (UI) is defined as one data bit-width, regardless of data rate. Eye width is represented in terms of the UI.

		Gradient Descent	Momentum	RMSProp
On Training Set	RMSE (UI)	0.006	0.006	0.008
	Maximum Relative Error (%)	7.9	8.1	8.7
On Validation Set	RMSE (UI)	0.008	0.008	0.01
	Maximum Relative Error (%)	10.6	9.3	9.5

gradient descent, momentum, and RMSProp methods [6], [7]. The RMSE on the training set with gradient descent method is 3.1 mV at the last iteration and 3.4 mV on the validation set. When the momentum method is used, the RMSE on the training set is reduced to 1.9 mV and 2.7 mV on the validation set. Figure 4 depicts the relative errors of the predicted eye heights on the validation set and the majority are below 3%. Another deep neural network (DNN) is trained to predict the eye width. The eye width is represented in terms of a unit interval (UI), which is defined as one data bit-width. This DNN has seven hidden layers of 10, 20, 20, 30, 20, 20, and 10 nodes, respectively. The training and validation sets have 509 and 203 examples, respectively. The batch size is changed to 15. The eye width varies from 0.21 to 0.37 UI in the data set. From Table II, the RMSE at the last iteration with momentum method is 0.006 UI on the training set and 0.008 UI on the validation set, and the maximum relative errors in both training and validation sets are less than 10%. It can be seen that the DNN models achieve good accuracies in predicting eye-diagram metrics. Figure 5 shows the convergence while training the DNN with the three optimizers. Through either adding a fraction of the updated vector from the previous step in the momentum method or adapting the learning rate in the RMSRprop method, faster convergence than the conventional gradient descent method can be achieved. It is worth mentioning that data standardization is applied to the training set such that an individual feature is transformed into a data set with zero mean and unit variance. The hyperbolic tangent is chosen as the activation function in both numerical examples.

IV. CONCLUSION

In this work, DNNs are trained to model high-speed channels for signal integrity analysis. The training and validation data sets are obtained from the simulation results in a previous design or at an earlier design stage. The proposed DNN models predict eye-diagram metrics through extrapolation with saved coefficients, which saves iterations of complex and

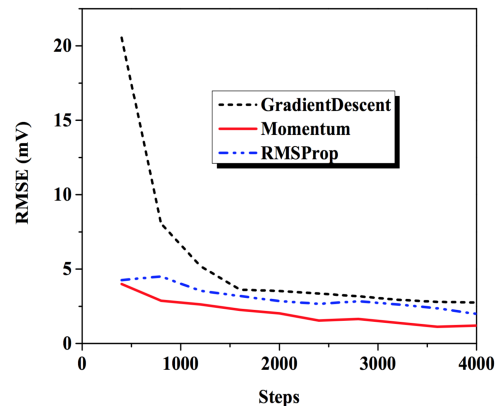


Fig. 5: The RMSE of eye height on the training set with three different optimizers.

computationally intensive simulations and enhances efficiency in signal integrity analysis. Numerical examples demonstrate that the DNN models achieve good accuracy in predicting eye height and width.

REFERENCES

- [1] J.-M. Jin, *The finite element method in electromagnetics*. John Wiley & Sons, 2015.
- [2] W.-D. Guo, J.-H. Lin, C.-M. Lin, T.-W. Huang, and R.-B. Wu, "Fast methodology for determining eye diagram characteristics of lossy transmission lines," *IEEE Transactions on Advanced Packaging*, vol. 32, no. 1, pp. 175–183, 2009.
- [3] J. Schmidhuber, "Deep learning in neural networks: an overview," *Neural networks*, vol. 61, pp. 85–117, 2015.
- [4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [5] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.
- [6] N. Qian, "On the momentum term in gradient descent learning algorithms," *Neural networks*, vol. 12, no. 1, pp. 145–151, 1999.
- [7] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.