Enhanced PML Based on the Long Short Term Memory Network for the FDTD Method

HE MING YAO, (Member, IEEE), AND LIJUN JIANG, (Fellow, IEEE)
Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong

Corresponding author: Lijun Jiang (jianglj@hku.hk)

ABSTRACT This paper proposes a new absorbing boundary condition (ABC) computation approach based on the deep learning technique. Benefited from the sequence dependence feature, the Long Short-Term Memory (LSTM) network is employed to replace the conventional perfectly matched layer (PML) ABC for the Finite-Difference Time-Domain (FDTD) solving process. The newly proposed LSTM based PML model is trained by the electromagnetic field data on the interface of the conventional PML. Different from the conventional PML, the newly proposed model only needs one cell layer as the boundary. Hence, the newly proposed method conveniently reduces both the algorithm's complexity and the area of computation domain of FDTD. Additionally, the newly proposed LSTM based PML model can achieve higher accuracy than the conventional artificial neural network (ANN) based PML, thanks to the sequence dependence feature of the LSTM networks. Numerical examples have illustrated the capability and the accuracy of the proposed LSTM model. The results illustrate that the new method can be compatibly embedded into the FDTD solving process with the high accuracy.

INDEX TERMS LSTM network, PML, deep learning, FDTD.

I. INTRODUCTION

Absorbing boundary conditions (ABCs) are widely utilized to truncate the computational area for unbounded problems during FDTD solving process [1]–[3]. Perfectly matched layer (PML), as the most widely-used ABC, has been developed in various application scenarios, thanks to its high tolerance to frequency and its excellent absorption capability over a broad scope of angles [4], [5]. Also, PML has been widely reported in various application of electromagnetic modeling, including convolutional PML (CPML) [6] and uniaxial PML (UPML) [7], [8]. But, the PML size has to be augmented to decrease the residue error for better absorption performance over wide range angles and broad frequency band. Hence, the expanded size leads to the increased computational complexity and computational domain.

Accompanied with the continuous development of computer science, machine learning (ML) approaches [9] have numerous successful application scenarios, including electromagnetic computation [10]–[13], electromagnetic inverse

The associate editor coordinating the review of this manuscript and approving it for publication was Mu-Yen Chen.

problems [14]–[17] and field-circuit cosimulations [18], [19]. The computing process within PML can be concluded as repeatively obtaining local field by using local and neighbouring field information in the previous and current step. This feature makes ML approaches adaptable to substitute for the computation process of conventional PML. The previous work has demonstrated the application of artificial neural network (ANN) model into replacing conventional PML [20]. But, the inevitably accumulative error in artificial neural network computation process greatly limits the usage of ANN based PML for FDTD solving process.

In this paper, we propose a new PML model based on the Long Short-Term Memory (LSTM) network. The LSTM network features with the ability of processing sequence-type data in the highly efficient and accurate mode, particularly for time-domain simulations [21], [22]. The electromagnetic (EM) field data on the interface between the first layer of conventional PML and the object domain are utilized to train the deep LSTM network model. The trained LSTM based PML model can replace conventional PML in multi-layer to reduce the computation complexity for FDTD solving process. The advantages of the newly proposed method
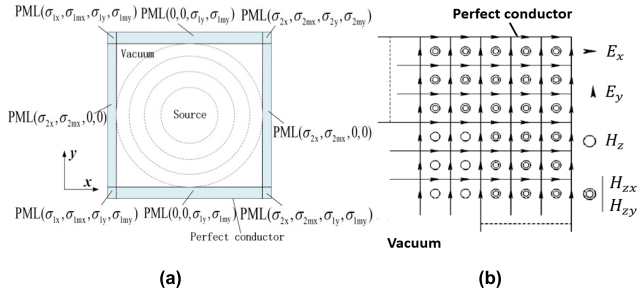
FIGURE 1. Formulation of conventional PML. (a) PML in 2D FDTD. (b) TEz PML grid.

are: (1) The LSTM based PML model reduces the computation domain and thereby the computation complexity for the FDTD ABC, because this new model only needs one-layer cell; (2) The newly proposed LSTM based PML is very convenient and flexible to be implemented in various FDTD application scenarios; (3) The accuracy of this proposed LSTM based PML model has been greatly increased (about 5dB), compared to the previous ANN based PML [20].

The structure of this paper is: the formulation of FDTD and the newly proposed LSTM model are described in Section II. Meanwhile, Section II also demonstrates the mechanism of integrating the proposed LSTM model to FDTD. Numerical benchmarks in the next section present the feasibility and accuracy of the proposed LSTM method, while the conclusion is offered in Section IV.

II. FORMULATIONS

A. FORMULATION OF CONVENTIONAL PML

The conventional PML medium in a 2D Transverse Electronic to the z direction (TE$_z$) wave case is shown in Fig. 1(a),

where the EM field in PML consists of four components (E_x, E_y, H_{zx}, H_{zy}) as described by the Yee cell [2], [23], specifically demonstrated by the four following equations:

where δ is the PML thickness, and ρ is the distance away from the interface. The conductivity in (1), as shown at the bottom of this page, is represented as $\sigma(\rho) = \sigma_{max}(\frac{\rho}{\delta})^p$ with p chosen as 1 or 2, while σ_{max} acts as the conductivity at the outermost layer of PML.

B. MECHANISM OF LSTM BASED PML

The LSTM network is a representative deep recurrent neural network (DRNN), which can learn long-term dependencies between time steps of sequence data [24]–[29]. LSTM is designed to overcome the inherent problems of RNNs, such as vanishing and exploding gradients [24]–[30]. Besides, LSTM block replaces the hidden layer mode for traditional "shallow" ANN. The LSTM block in Fig.2(b) can "store", "write" and "read" data via gates that open and close [26]–[30], which makes it matching the nonlinearities of real sequence data better than traditional ANN and RNN [24]–[30]. Considering the computation process within conventional PML, each cell in conventional PML repeatively computes local field based on its local and neighbouring fields at the previous and current time step [2], [5], [23]. Therefore, this computing process could be referred as the nonlinear sequence-dependence computation and it can be handled by LSTM model. By opening and closing gates, LSTM cells can 'forget' the potential useless information in previous time steps, 'update' state and 'improve' the output for the current time step in FDTD solving process.

The core component of an LSTM network is the LSTM layer, where long-term dependencies between time steps of

$$
\begin{cases}
E_x^{n+1/2}\left(i, j+\frac{1}{2}\right) = \exp\left[-\sigma_y\left(j+\frac{1}{2}\right)\frac{\Delta t}{\varepsilon_0}\right] E_x^{n-1/2}\left(i, j+\frac{1}{2}\right) \\
\qquad - \dfrac{1 - \exp\left[-\sigma_y\left(j+\frac{1}{2}\right)\frac{\Delta t}{\varepsilon_0}\right]}{\Delta y \sigma_y\left(j+\frac{1}{2}\right)}\left[H_z^n(i,j) - H_z^n(i, j+1)\right] \\
E_y^{n+1/2}\left(i+\frac{1}{2}, j\right) = \exp\left[-\sigma_x\left(i+\frac{1}{2}\right)\frac{\Delta t}{\varepsilon_0}\right] E_y^{n-1/2}\left(i+\frac{1}{2}, j\right) \\
\qquad - \dfrac{1 - \exp\left[-\sigma_x\left(i+\frac{1}{2}\right)\frac{\Delta t}{\varepsilon_0}\right]}{\Delta x \sigma_y\left(i+\frac{1}{2}\right)}\left[H_z^n(i+1, j) - H_z^n(i, j)\right] \\
H_{zx}^{n+1}(i,j) = \exp\left[-\sigma_{mx}(i)\frac{\Delta t}{\mu_0}\right] H_{zx}^n(i,j) - \dfrac{1 - \exp\left[-\sigma_{mx}(i)\frac{\Delta t}{\mu_0}\right]}{\Delta x \sigma_{mx}(i)} \\
\qquad \times \left[E_y^{n+1/2}\left(i+\frac{1}{2}, j\right) - E_y^{n+1/2}\left(i-\frac{1}{2}, j\right)\right] \\
H_{zy}^{n+1}(i,j) = \exp\left[-\sigma_{my}(j)\frac{\Delta t}{\mu_0}\right] H_{zy}^n(i,j) - \dfrac{1 - \exp\left[-\sigma_{my}(j)\frac{\Delta t}{\mu_0}\right]}{\Delta x \sigma_{mx}(j)} \\
\qquad \times \left[E_x^{n+1/2}\left(i, j+\frac{1}{2}\right) - E_x^{n+1/2}\left(i, j-\frac{1}{2}\right)\right]
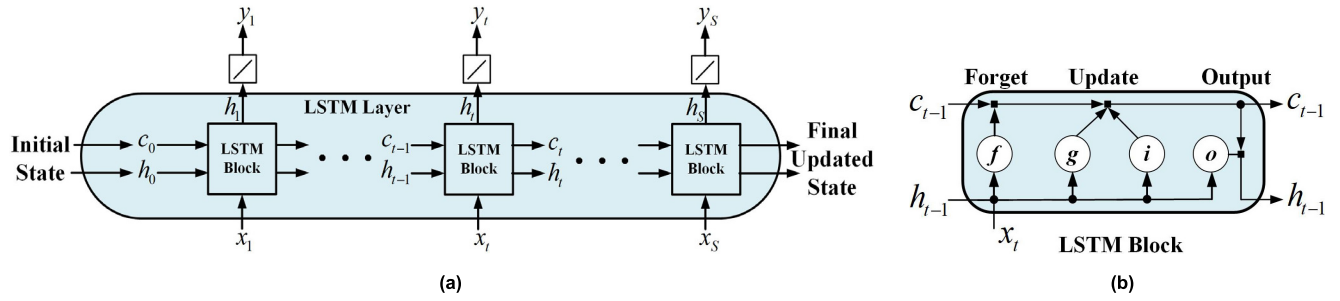\end{cases}
\tag{1}
$$

FIGURE 2. (a) Architecture of LSTM layer. (b) LSTM block [26]–[30].

sequence data are learnt, as is presented in Fig. 2. The time series or sequence data x_t are input into this network by the sequence input layer. In Fig.2, the flow of a time series x_t of length S is input through the LSTM layer, where h_t is known as the hidden state and c_t denotes the cell state. The output y_t is ultimately obtained by doing linear transformation to h_t. At the time step t, the LSTM block makes use of the previous state (c_{t-1}, h_{t-1}) and the current input x_t, and then updates h_t and c_t for the current time step. The hidden state and the cell state involves information learned from the previous time steps. Thus, the LSTM layer can absorb or delete information by using the cell state at each time step. These updates are totally controlled by gates of the LSTM layer, presented in Fig.2(b).

The mechanism of LSTM is based on the following equations (2)-(8). The specific computation process within equations (2)-(8) is as follows: the initial state and the sequence in the first time step is used to calculate the first output and update the cell states. Then, for each time step t, the LSTM block conducts the same operation, utilizing the state and the input sequence to calculate the output and cell state. At the time step t, the Fig.2(b) illustrates how the gates forget, update, and output the hidden and the cell states in the LSTM block. The layer state includes the hidden state and the cell state. The important sequence information is involved in the cell state. For each time step t, the layer 'store' or 'forget' information from the cell state [24]–[30], as described in the equations (2)-(8).

$$i_t = \sigma_g \left(W_i x_t + R_i h_{t-1} + b_i \right) \tag{2}$$

$$g_t = \sigma_c \left(W_g x_t + R_g h_{t-1} + b_g \right) \tag{3}$$

$$f_t = \sigma_g \left(W_f x_t + R_f h_{t-1} + b_f \right) \tag{4}$$

$$o_t = \sigma_g \left(W_o x_t + R_o h_{t-1} + b_o \right) \tag{5}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{6}$$

$$h_t = o_t \odot \sigma_c \left(c_t \right) \tag{7}$$

$$y_t = U h_t + b_y \tag{8}$$

The learnable weights of LSTM layer are output weight U, the input weights $W = \{W_i, W_f, W_g, W_o\}$, the recurrent weights $R = \{R_i, R_f, R_g, R_o\}$ and the bias $b = \{b_i, b_f, b_g, b_o, b_y\}$. i_t, g_t, f_t and o_t respectively represent the input gate, forget gate, cell candidate, and output gate at time
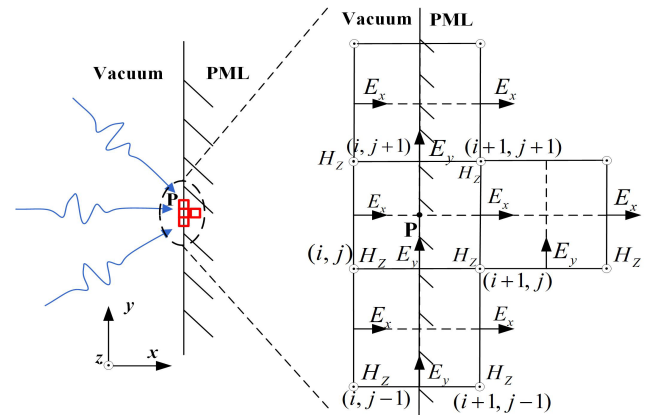

FIGURE 3. Mechanism of collecting training data for LSTM based PML in the FDTD scenario for TE$_z$ wave.

step t, while $\odot$ denotes the Hadamard product. Plus, the state activation function σ_c is denoted by the hyperbolic tangent function, and the gate activation function σ_g is chosen as the sigmoid function [24]–[30].

The application of the proposed LSTM PML consists of two stages: the off-line training stage and the online prediction stage. In the offline training stage, the EM field data on the interface between the first layer of conventional PML and the object domain are used to train the deep LSTM network model, so that the prior information of FDTD solving process in the PML is integrated into the trained model. In the online stage, the trained LSTM PML is integrated into the FDTD solving process to do the further field prediction with satisfactory accuracy and efficiency.

The specific offline training process of the proposed LSTM model is presented in the next section. For the proposed LSTM model, the dimension of input and output are 12 and 3, respectively, to match the field information chosen as the training data. Based on the training dataset, the learnable weights W, R, U and b could be ultimately determined by various training algorithms. In this paper, Adaptive Moment Estimation (Adam) optimizer is chosen to optimize the half-mean-squared-error loss function, as reported in [31]–[35]. The training is done by 'sequence-to-sequence' method for LSTM [26]–[30].

Fig. 3 presents the mechanism of collecting data for training the LSTM based PML model. While the node P is on the
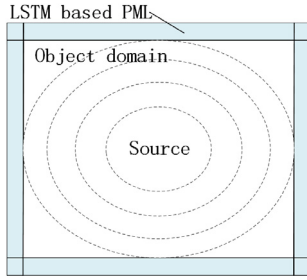
FIGURE 4. Integration of LSTM based PML model to FDTD computing process.

first layer of conventional PML, the cells near P receive waves from different directions in Fig.3. Based on the wave in each direction, the local and neighbouring field data of the node P in the continuous time sequence are organized to form the training data. In Fig. 3, TE_z wave is utilized as the example, where we set input data and output data as x_{ij}^t and y_{ij}^t:

$$
\begin{cases}
x_{ij}^t = [E_x^t\left(i, j+1/2\right), E_x^t\left(i, j-1/2\right), E_x^t\left(i, j-3/2\right), \\
\qquad E_x^t\left(i+1, j-1/2\right), E_y^t\left(i+1/2, j+1\right), \\
\qquad E_y^t\left(i+1/2, j\right), E_y^t\left(i+1/2, j-1\right), E_y^t\left(i+3/2, j\right) \\
\qquad H_z^t\left(i, j+1\right), H_z^t\left(i, j\right), H_z^t\left(i, j-1\right), H_z^t\left(i+1, j\right)]^T \\
y_{ij}^t = [H_z^{t+1}\left(i+1, j\right), E_x^{t+1}\left(i+1, j-1/2\right), \\
\qquad E_y^{t+1}\left(i+3/2, j\right)]^T
\end{cases}
$$

To obtain the training data, we collect chunks of data group $\left\{x_{ij}^{t_1}, \ldots, x_{ij}^{t_S}\right\}$ and $\left\{y_{ij}^{t_1}, \ldots, y_{ij}^{t_S}\right\}$ with features of field on the interface of PML. Subsequently, based on these $\left\{x_{ij}^{t_1}, \ldots, x_{ij}^{t_S}\right\}$ and $\left\{y_{ij}^{t_1}, \ldots, y_{ij}^{t_S}\right\}$, LSTM based PML model can be trained. Here, we emphasize that all these $\left\{x_{ij}^{t_1}, \ldots, x_{ij}^{t_S}\right\}$ and $\left\{y_{ij}^{t_1}, \ldots, y_{ij}^{t_S}\right\}$ groups are collected on the interface of conventional PML with 40-layer thickness, which has been widely accepted to be sufficient to effectively absorb the electromagnetic waves and delete reflection between layer interfaces [2]–[5].

Several issues are still required to be discussed for creating our LSTM model, including number of hidden unit and sequence length of LSTM.

1) ACTIVATION FUNCTION
using the sigmoid and hyperbolic tangent activation functions makes gradient vanishing in conventional RNN, especially when the training algorithm Back-Propagation Through Time (BPTT) is utilized [24]–[30]. However, this problem is resolved by the network structure of a LSTM, thanks to the various gates and the memory cell [26]–[30].

2) SEQUENCE LENGTH
according to [27]–[33], the increase of sequence length S for training data set cannot only help the LSTM model improve accuracy, but also makes training converge faster. However, when increasing S up to a certain threshold, neither accuracy nor convergence can be improved any more [31]–[33].

In this case, more training computations are required due to the increase of S, which unavoidably degrades computation efficiency [27]–[33]. Ultimately, we set the sequence length as $S = 15$ for the proposed LSTM based PML model, which has been widely chosen in the numerous physical-based cases [31]–[33], [36]–[38].

3) NUMBER OF HIDDEN UNIT
in most cases, the determination of the exact number of hidden unit for h_t is still an important research area for deep learning [18]–[20], [39]–[40]. In this paper, as the most commonly used strategy for deep learning, the trial-and-error process is taken to ensure the number of hidden unit [18]–[20]. While various number of hidden units can be tried, the expected values are the smaller ones resulting into smaller training errors and less computation. Taking the precision and the number of parameters of the proposed model into account, we ultimately set the number of hidden unit as 20.

Therefore, the final LSTM based model is represented in the equations (2)-(8). This trained LSTM model with the prior information of FDTD solving process in the PML can then be integrated into the FDTD solving process to do the further field prediction with satisfactory accuracy and efficiency. Based on the resultant LSTM model, $y_{ij}^{t_S}$ can be predicted by $x_{ij}^{t_S}$ and the field information on the PML interface at t_{S+1} can be further obtained. Meanwhile, the states of each grid of one-cell LSTM based PML at t_{S+1} are computed and stored to be used in the next time step. The index i and j in this section represent that of the grid for the LSTM based PML, which act as the node P presented in Fig. 3. Then, using $y_{ij}^{t_S}$, we can calculate field in the object domain at t_{S+1} and then gain $x_{ij}^{t_{S+1}}$, by standard FDTD process. The further predictions are done for boundaries at t_{S+2}, and then further time steps. By exactly repeating this process, the proposed LSTM model can replace the entire conventional PML and can be embedded to the FDTD solving process. Hence, for each grid on the LSTM PML, the field information can always be predicted by using field information on local and neighbouring grids. The program workflow is summarized as:

III. NUMERICAL EXAMPLES
We use several representative numerical benchmarks to validate the performance of this newly proposed LSTM based PML model. The conventional 1-cell PML, the conventional 5-cell PML and the HTBF based PML [20] are utilized to compare with the proposed LSTM based PML.

A. POINT SOURCE EXCITATION IN A SQUARE REGION
In the first benchmark, LSTM based PML is employed in 2D square area in TE_z wave, presented in Fig. 5(a). While the chosen time step is $\Delta t = 1ps$, a 15mm×15mm square domain is discretized as $\Delta x = \Delta y = 1mm$ cells. Fig. 5(b) shows the conventional PML scenario, where the same square area involves a 5-cell conventional PML at each boundary.

Algorithm 1 Embedding LSTM based PML into FDTD

Model Training

1) Collect groups of $\left\{x_{ij}^{t_1}, \ldots, x_{ij}^{t_S}\right\}$, $\left\{y_{ij}^{t_1}, \ldots, y_{ij}^{t_S}\right\}$ as Fig. 3.
2) Train LSTM based PML model by using the data set in the
   above step, as is represented in (2)-(8).

Model Embedding

Initialization:

1) Initialize H^{t_0} and E^{t_0} in the object domain and one-cell LSTM PML as shown in Fig. 4.
2) Do zero-padding for c_t and h_t with $t < t_0$ for each cell on LSTM PML.

kth Iteration:

3) Update H^{t_k} and E^{t_k} at one-cell LSTM based PML by using $H^{t_{k-1}}$ and $E^{t_{k-1}}$ in object domain, $H^{t_{k-1}}$ and $E^{t_{k-1}}$ in one-cell LSTM based PML, and $c_{t_{k-1}}$ and $h_{t_{k-1}}$ on each cell on LSTM PML
4) Update c_{t_k} and h_{t_k} for each cell on LSTM PML
5) Update H^{t_k} and E^{t_k} in object domain by using H^{t_k} and E^{t_k} at one-cell LSTM based PML with standard FDTD process.
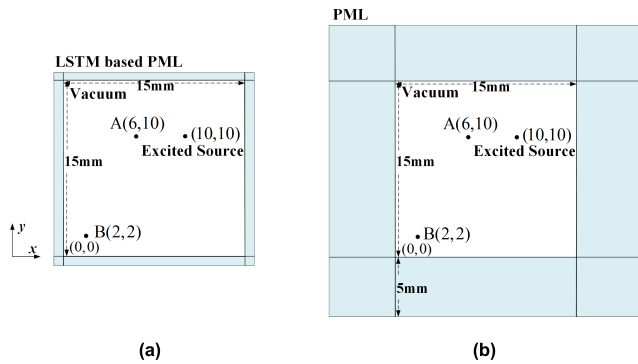

FIGURE 5. The 15mm×15mm square area for FDTD scenario with two probes and one excited source. (a) LSTM based PML. (b) 5-cell conventional PML.

But, the proposed LSTM based PML only needs one cell at each edge in Fig.5(a). At the position $x_0 = y_0 = 10$ mm, there is a sinusoidal magnetic current source in z direction, formulated as the following expression:

$$H_z(x_0, y_0, t) = A \cdot \sin[2\pi f_c(t - t_0)] \qquad (9)$$

where $A = 1$, $f_c = 10$GHz, $t_0 = 0$.

The Magnetic fields (H_z) at A (6,10) and B (2, 2) are probed. Normally, 100 time steps are seen as the steady state respond [1]–[3]. We use the following equation to define the relative error at each point:

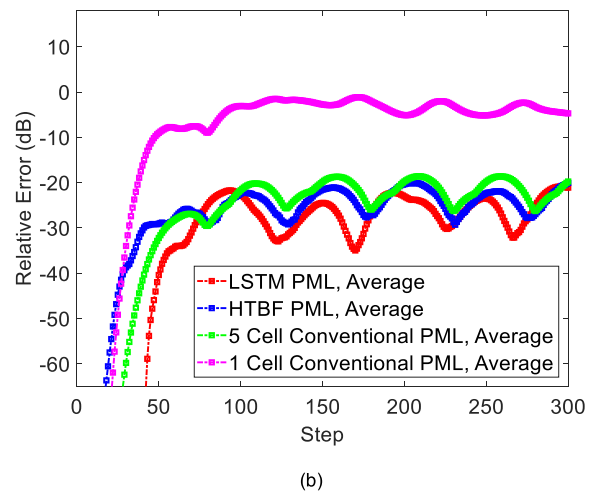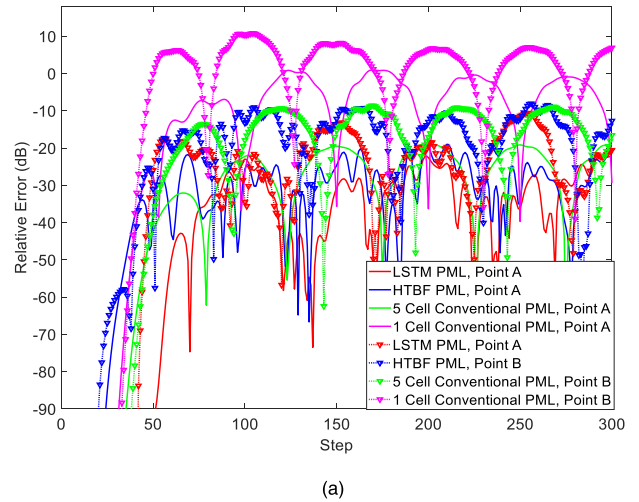$$e(t) = \frac{\left|H_z(t) - H_z^{ref}(t)\right|}{\left|H_z^{ref,max}\right|} \qquad (10)$$

FIGURE 6. Relative errors comparison among four PML methods. (a) Relative error at Point A and B. (b) Average of relative error on the whole square.

where $H_z(t)$ is the magnetic fields probed on the measured points in this 15mm×15mm square, while $H_z^{ref}(t)$ is the reference magnetic fields in a 215mm×215mm square on accurately the same relative positions from the source. $H_z^{ref,max}$ denotes the maximum amplitude of the reference field probed at the local points.

In this example, 5000 groups of training data is utilized to train our LSTM model in the offline training stage. We randomly choose 80% of the data groups to do training, 10% for validating, and another 10% for testing. The initial learning rate is 0.001, while the min-max normalization is adopted for better training [31]–[33], [41]. The proposed model is implemented in the Deep Learning Toolbox in Matlab 2019a [42]. The training of this LSTM model takes 34.83s CPU time. The performance of our LSTM model in the online application stage is demonstrated in Fig. 6, where the relative errors of LSTM PML model at Point A and B are compared both with the 5-cell and 1-cell conventional PML and with the previously proposed HTBF based PML [20]. As shown in Fig. 6(a), the maximum of relative error from LSTM based
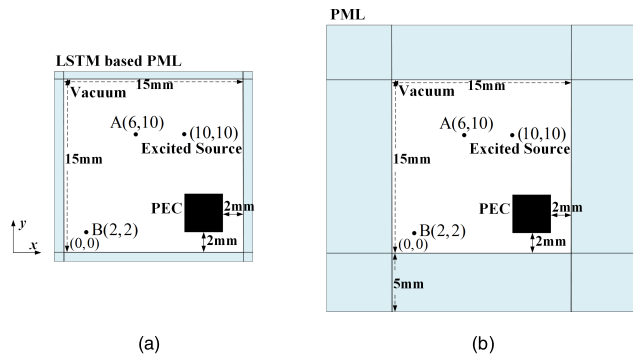
FIGURE 7. The 15mm×15mm square area for FDTD scenario with the PEC, two probes and one excited source. (a) LSTM based PML. (b) 5-cell conventional PML.

PML are about −30dB at Point A, while that maximum for 1-cell conventional PML, 5-cell conventional PML and HTBF based PML arrive at 0dB, −20dB and −25dB, respectively. Moreover, it can be seen that the maximum of relative errors at Point B for four methods are respectively 5dB for the 1-cell conventional PML, −10dB for the 5-cell conventional PML, −15dB for HTBF based PML and −20dB for LSTM based PML. Besides, Fig. 6(b) presents that the average of relative error is −20dB for both 5-cell conventional PML and HTBF based PML model, while it is below −25dB for our novel LSTM model. However, that average of 1-cell conventional PML is almost 0dB. From the error analysis above, we can conclude that 1-cell conventional PML cannot be offer any meaningful information for FDTD solving process, while the LSTM based PML provides the best accuracy for this case.

According to Fig.6, our LSTM based PML is evidently better than HTBF based PML and conventional PML. Besides, only one cell is used to finish absorbing for the LSTM based PML. Furthermore, CPU memory can be considerably saved by the proposed LSTM model because the FDTD computation process repeatedly uses the same model. The CPU time for the implementation of four methods are 0.347s for the 1-cell conventional PML, 0.486s for HTBF PML model, 0.624s for LSTM model and 0.715s for 5-cell conventional PML.

B. SCATTERING BY PEC OBJECT

While the second benchmark takes all similar setups with the Section III A, a PEC square with the size of 3mm×3mm is added into the 15mm×15mm square area.

In this benchmark, the same model trained in Section III A is employed to demonstrate the generalization of the proposed LSTM method. According to Fig. 8(a), the proposed LSTM based PML has better performance in its online application stage. The results present that the maximum of relative error of HTBF based PML and LSTM based PML and 5-cell conventional PML are all around −20dB at A point, while that relative error is nearly 0dB for the 1-cell conventional PML. Furthermore, the maximum of relative errors of HTBF based
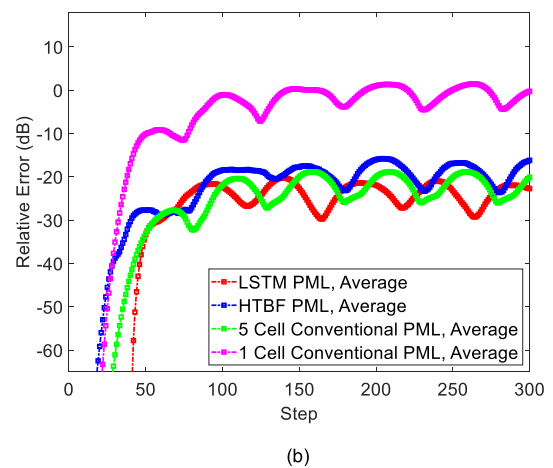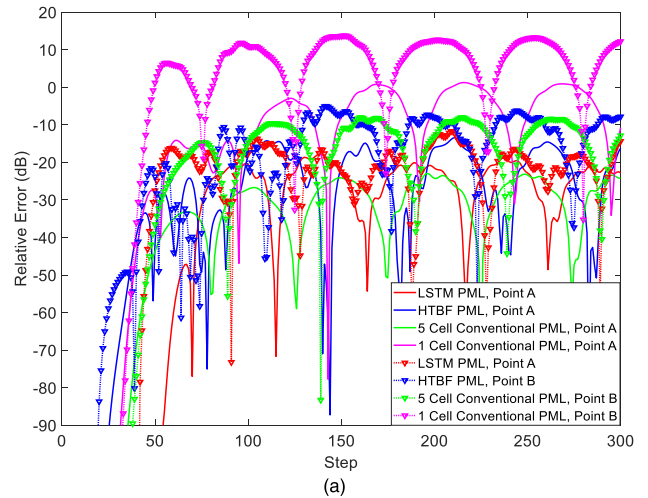

FIGURE 8. Relative errors comparison among four PML methods. (a) Relative error at Point A and B. (b) Average of relative error on the whole square.

PML and 5-cell conventional PML are both around −10dB PML at Point B, while that of LSTM based PML is about −18dB. But, as the worst case, the maximum of relative error of 1-cell conventional PML can overcome 5dB at B point. Moreover, Fig. 8(b) presents the average of relative errors of the four methods. The average relative errors are about −20dB for HTBF based PML and 5-cell conventional PML, and 0dB for 1-cell conventional PML, while that average is below −25dB for the proposed LSTM model. The result of the error analysis above is the same as the previous section: 1-cell conventional PML cannot be used for FDTD, while the LSTM based PML provides the best accuracy.

From the comparison above, we can conclude that LSTM based PML is evidently better in terms of accuracy, while the accuracy of 5-cell conventional PML is approximated to that of the HTBF based PML. The CPU time for the implementation of four methods are 0.397s for the 1-cell conventional PML, 0.438s for HTBF PML model, 0.601s for LSTM PML model, and 0.833s for 5-cell conventional PML.

IV. CONCLUSION

This work integrates deep learning approach into the absorbing boundary condition for the FDTD. The newly proposed LSTM based PML substitutes for the conventional PML ABC. Unlike the conventional PML, only a one-cell boundary layer is required by this newly proposed LSTM based PML model to absorb the propagating wave. Hence, it considerably reduces the size of the computation domain. Compared with the conventional PML and conventional ANN based PML, the new method provides both good accuracy and low computation complexity simultaneously. Numerical examples demonstrate that the newly proposed LSTM based PML model can successfully replace the conventional PML to be flexibly embedded to the FDTD solving process. The newly proposed method opens a new PML algorithm path to the deep learning regime.

REFERENCES

[1] G. Mur, "Absorbing boundary conditions for the finite-difference approximation of the time-domain electromagnetic-field equations," *IEEE Trans. Electromagn. Compat.*, vol. EMC-23, no. 4, pp. 377–382, Nov. 1981.

[2] D. M. Sullivan, *Electromagnetic Simulation Using FDTD Method*. Piscataway, NJ, USA: Wiley, 2013.

[3] K. Fujita, "Complex-frequency shifted PML formulation for the 3-D MNL-FDTD method," *IEEE Microw. Wireless Compon. Lett.*, vol. 26, no. 9, pp. 651–653, Sep. 2016.

[4] J.-P. Berenger, "Improved PML for the FDTD solution of wave-structure interaction problems," *IEEE Trans. Antennas Propag.*, vol. 45, no. 3, pp. 466–473, Mar. 1997.

[5] A. Taflove, A. Oskooi, S. G. Johnson, *Advances in FDTD Computational Electrodynamics: Photonics and Nanotechnology*. Boston, MA, USA: Artech House, 2013.

[6] X.-H. Wang, W.-Y. Yin, Y.-Q. Yu, Z. Chen, J. Wang, and Y. Guo, "A convolutional perfect matched layer (CPML) for one-step leapfrog ADI-FDTD method and its applications to EMC problems," *IEEE Trans. Electromagn. Compat.*, vol. 54, no. 5, pp. 1066–1076, Oct. 2012.

[7] A. Ping Zhao, "Uniaxial perfectly matched layer media for an unconditionally stable 3-D ADI-FD-TD method," *IEEE Microw. Wireless Compon. Lett.*, vol. 12, no. 12, pp. 497–499, Dec. 2002.

[8] L. Bernard, R. R. Torrado, and L. Pichon, "Efficient implementation of the UPML in the generalized finite-difference time-domain method," *IEEE Trans. Magn.*, vol. 46, no. 8, pp. 3492–3495, Aug. 2010.

[9] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, Aug. 2006.

[10] H. Yao, L. Jiang, and Y. Qin, "Machine learning based method of moments (ML-MoM)," in *Proc. IEEE Int. Symp. Antennas Propag. USNC/URSI Nat. Radio Sci. Meeting*, Jul. 2017, pp. 973–974.

[11] T. Shan, W. Tang, X. W. Dang, M. K. Li, F. Yang, S. H. Xu, and J. Wu, "Study on a Poisson's equation solver based on deep learning technique," 2017, *arXiv:1712.05559*. [Online]. Available: https://arxiv.org/abs/1712.05559

[12] H. Yao and L. Jiang, "Machine learning based neural network solving methods for the FDTD method," in *Proc. IEEE Int. Symp. Antennas Propag. USNC/URSI Nat. Radio Sci. Meeting*, Jul. 2018, pp. 2321–2322.

[13] H. M. Yao, M. Li, and L. Jiang, "Applying deep learning approach to the far-field subwavelength imaging based on near-field resonant metalens at microwave frequencies," *IEEE Access*, vol. 7, pp. 63801–63808, 2019.

[14] R. Guo, X. Song, M. Li, F. Yang, S. Xu, and A. Abubakar, "Supervised descent learning technique for 2-D microwave imaging," *IEEE Trans. Antennas Propag.*, vol. 67, no. 5, pp. 3550–3554, May 2019.

[15] H. M. Yao, W. E. I. Sha, and L. J. Jiang, "Applying convolutional neural networks for the source reconstruction," *Prog. Electromagn. Res. M*, vol. 76, pp. 91–99, 2018, doi: 10.2528/pierm18082907.

[16] Z. Wei and X. Chen, "Deep-learning schemes for full-wave nonlinear inverse scattering problems," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 1849–1860, Apr. 2019, doi: 10.1109/tgrs.2018.2869221.

[17] H. M. Yao, W. E. I. Sha, and L. Jiang, "Two-step enhanced deep learning approach for electromagnetic inverse scattering problems," *IEEE Antennas Wireless Propag. Lett.*, vol. 18, no. 11, pp. 2254–2258, Nov. 2019, doi: 10.1109/lawp.2019.2925578.

[18] H. H. Zhang, L. J. Jiang, and H. M. Yao, "Embedding the behavior macro-model into TDIE for transient field-circuit simulations," *IEEE Trans. Antennas Propag.*, vol. 64, no. 7, pp. 3233–3238, Jul. 2016.

[19] H. H. Zhang, L. J. Jiang, H. M. Yao, and Y. Zhang, "Transient heterogeneous electromagnetic simulation with DGTD and behavioral macro-model," *IEEE Trans. Electromagn. Compat.*, vol. 59, no. 4, pp. 1152–1160, Aug. 2017.

[20] H. M. Yao and L. Jiang, "Machine-learning-based PML for the FDTD method," *IEEE Antennas Wireless Propag. Lett.*, vol. 18, no. 1, pp. 192–196, Jan. 2019.

[21] D. Hsu, "Multi-period time series modeling with sparsity via Bayesian variational inference," 2017, *arXiv:1707.00666*. [Online]. Available: https://arxiv.org/abs/1707.00666

[22] Y. Wang, "A new concept using LSTM neural networks for dynamic system identification," in *Proc. Amer. Control Conf. (ACC)*, May 2017, pp. 5324–5329.

[23] K. Yee, "Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media," *IEEE Trans. Antennas Propag.*, vol. 14, no. 3, pp. 302–307, May 1966.

[24] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics 7th Int. Joint Conf. Natural Lang. Process.*, vol. 1, Jul. 2015, pp. 1556–1566.

[25] J. Cheng, L. Dong, and M. Lapata, "Long short-term memory-networks for machine reading," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 551–561.

[26] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[27] M. Sundermeyer, R. Schlüter, and H. Ney, "LSTM neural networks for language modeling," in *Proc. INTERSPEECH*, 2012, pp. 194–197.

[28] C. Olah. (2015). *Understanding LSTM Networks*. [Online]. Available: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[29] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," in *Proc. 9th Int. Conf. Artif. Neural Netw. (ICANN)*, vol. 2, Sep. 1999, pp. 850–855.

[30] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112.

[31] T. Nguyen, T. Lu, J. Sun, Q. Le, K. We, and J. Schut-Aine, "Transient simulation for high-speed channels with recurrent neural network," in *Proc. IEEE 27th Conf. Electron. Perform. Electron. Packag. Syst. (EPEPS)*, San Jose, CA, USA, Oct. 2018, pp. 303–305.

[32] Y. Choi, S. Ryu, K. Park, and H. Kim, "Machine learning-based lithium-ion battery capacity estimation exploiting multi-channel charging profiles," *IEEE Access*, vol. 7, pp. 75143–75152, 2019.

[33] T. Nguyen, T. Lu, K. Wu, and J. Schutt-Aine, "Fast transient simulation of high-speed channels using recurrent neural network," 2019, *arXiv:1902.02627*. [Online]. Available: https://arxiv.org/abs/1902.02627

[34] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–41.

[35] V. Pham, T. Bluche, C. Kermorvant, J. Louradour, "Dropout improves recurrent neural networks for handwriting recognition," Nov. 2013, *arXiv:1312.4569*. [Online]. Available: https://arxiv.org/abs/1312.4569

[36] G. Liu, Y. Xu, Z. He, Y. Rao, J. Xia, and L. Fan, "Deep learning-based channel prediction for edge computing networks toward intelligent connected vehicles," *IEEE Access*, vol. 7, pp. 114487–114495, 2019.

[37] I. Srivani, G. Siva Vara Prasad, and D. Venkata Ratnam, "A deep learning-based approach to forecast ionospheric delays for GPS signals," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 8, pp. 1180–1184, Aug. 2019.

[38] X. Lu, "Memory-controlled deep LSTM neural network post-equalizer used in high-speed PAM VLC system," *Opt. Express*, vol. 27, no. 5, pp. 7822–7833, Feb. 2019.

[39] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[40] A. Graves, A. Mohamed, and G. E. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, May 2013, pp. 6645–6649.

[41] A. Jain, K. Nandakumar, and A. Ross, "Score normalization in multimodal biometric systems," *Pattern Recognit.*, vol. 38, no. 12, pp. 2270–2285, Dec. 2005.

[42] P. Kim, *MATLAB Deep Learning*. New York, NY, USA: Apress, 2017.

HE MING YAO (Member, IEEE) is currently pursuing the Ph.D. degree in electrical and electronic engineering with The University of Hong Kong, Hong Kong. His research interests include CEM, EMC/EMI, machine learning, and advanced materials.

LIJUN JIANG (Fellow, IEEE) received the B.S. degree in electrical engineering from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 1993, the M.S. degree from Tsinghua University, Beijing, in 1996, and the Ph.D. degree from the University of Illinois at Urbana–Champaign, Champaign, IL, USA, in 2004. From 1996 to 1999, he was an Application Engineer with Hewlett-Packard Company. Since 2004, he has been a Postdoctoral Researcher, a Research Staff Member, and a Senior Engineer with the IBM T. J. Watson Research Center, Yorktown Heights, NY, USA. Since 2009, he has also been an Associate Professor with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong. Since 2013, he has also been a Senior Visiting Professor with Tsinghua University. He has served as a Scientific Consultant with Hong Kong Applied Science and Technology Research Institute Company Ltd., from 2010 to 2011. He has been serving as a Panelist on the Expert Review Panel of Hong Kong Research and Development Center for logistics and supply chain management enabling technologies, since 2013. He was the Semiconductor Research Cooperation (SRC) Industrial Liaison for several academic projects. He has been involved collaboratively with many international researchers. He is an IEEE AP-S Member, an IEEE MTT-S Member, an IEEE EMC-S Member, an ACES Member, and a member of the Chinese Computational Electromagnetics Society. He has been an Elected TPC Member of the IEEE Electrical Design of Advanced Packaging and Systems Symposium (EDAPS), since 2010, and the IEEE Electrical Performance of Electronic Packaging, since 2014. He was a TPC Member of the 2013 IEEE International Conference on Microwave Technology and Computational Electromagnetics. He has been a TC-9 and a TC-10 Member of the IEEE Electromagnetic Compatibility Society, since 2011. He was a Scientific Committee Member of the 2010 IEEE Simulation and Modeling of Emerging Electronics, the Special Session Organizer of the IEEE EDAPS, the IEEE Electromagnetic Compatibility, Applied Computational Electromagnetics Society, Asia-Pacific Radio Science Conference, and Progress in Electromagnetics Research Symposium, and a Co-Organizer of the HKU Computational Science and Engineering Workshops, from 2010 to 2012. He received the IEEE MTT Graduate Fellowship Award, in 2003, and the Y. T. Lo Outstanding Research Award, in 2004. He was the Session Chair of many international conferences, the TPC Chair of the 7th International Conference on Nanophotonics/the 3rd Conference on Advances in Optoelectronics and Micro/Nano Optics, the TPC Co-Chair of the 12th International Workshop on Finite Elements for Microwave Engineering and the 2013 International Workshop on Pulsed Electromagnetic Field at Delft, The Netherlands, and the General Chair of the 2014 IEEE 14th HK AP/MTT Postgraduate Conference. He was an Associate Guest Editor of the PROCEEDINGS OF THE IEEE special issue, from 2011 to 2012. He is also an Associate Editor of the IEEE TRANSACTIONS ON ANTENNAS AND PROPAGATION and *Progress in Electromagnetics Research*. He also serves as a Reviewer for the IEEE TRANSACTIONS on several topics and other primary electromagnetics and microwave related journals.

• • •