A Space–Time Neural Network for Analysis of Stress Evolution Under DC Current Stressing

Tianshu Hou\textsuperscript{1}, Ngai Wong\textsuperscript{1}, \textit{Senior Member, IEEE}, Quan Chen\textsuperscript{1}, \textit{Member, IEEE}, Zhigang Ji\textsuperscript{1}, and Hai-Bao Chen\textsuperscript{2}, \textit{Member, IEEE}

Abstract—The electromigration (EM)-induced reliability issues in very large-scale integration (VLSI) circuits have attracted increased attention due to the continuous technology scaling. Traditional EM models often lead to overly pessimistic predictions incompatible with the shrinking design margin in future technology nodes. Motivated by the latest success of neural networks in solving differential equations in physical problems, we propose a novel mesh-free model to compute EM-induced stress evolution in VLSI circuits. The model utilizes a specifically crafted space–time physics-informed neural network (STPINN) and as the solver for EM analysis. By coupling the physics-based EM analysis with dynamic temperature incorporating Joule heating and via effect, we can observe stress evolution along multisegment interconnect trees under constant, time-dependent, and space–time-dependent temperature during the void nucleation phase. The proposed STPINN method obviates the time discretization and meshing required in conventional numerical stress evolution analysis and offers significant computational savings. Numerical comparison with competing schemes demonstrates a 2x–52x speedup with a satisfactory accuracy.

Index Terms—Electromigration, machine learning, space–time aware, stress evolution, via effect.

I. INTRODUCTION

ELECTROMIGRATION (EM) has become a major concern for nanometer very large-scale integration (VLSI) designs due to the shrinking feature sizes and increased current density of copper damascene wire interconnects, especially for power grids carrying large unidirectional currents [1]. Consequently, EM verification is important in chip design sign-off. The well-known empirical Black’s model [2] estimates the mean time to failure (MTTF) of single metal wire based on the equation

\begin{equation}
\text{MTTF} = A j^{-n} \exp\left(\frac{E_a}{kT}\right)
\end{equation}

where \( j, T, k, E_a, n, A \) are the current density, temperature, Boltzmann’s constant, EM activation energy, current density exponent, and empirical constant, respectively. It should be noted that \( n \) and \( E_a \) have typical values for copper metallization by enough stress tests. According to highly accelerated tests using Joule heating, Black’s equation is employed for extrapolation to operating conditions, while both temperature and current density in stress conditions cannot be varied independently [3]. Besides, Black’s model and Blech’s effect (immortal segment filter) mainly target single wires, which may lead to significant errors and excessive design guard band in real designs [4]–[6]. In practice, VLSI circuits are composed of multiple interleaved interconnects, whose stress evolution is interactive among wires and must be considered altogether [7].

Physics-based models have been proposed in EM-induced stress evolution assessment in recent years [8]–[12]. These EM models focus on hydrostatic stress diffusion kinetics in continuously connected confined metal wires governed by coupled partial differential equations (PDEs) and on the accurate failure time estimation through critical stress. However, the mesh-based numerical methods, such as the finite difference method (FDM) and the finite element method (FEM) are constrained by the discretization of space and time, which increase the number of variables to solve PDEs resulting from complex on-chip interconnect topologies [13]. A voltage-based EM modeling and immortality check technique for general interconnect trees was proposed in [14]. In this work, if the largest stress of tree nodes is less than the critical stress, the tree is considered to be immortal. Although it is helpful for a fast EM immortality check, the predicted failure time of wires cannot be obtained. In [15] and [8], compact models were developed to extend stress diffusion on the single metal wire to the multisegment interconnect structure through known steady-state stress distribution based on Korhonen’s equation [16]. In [17], the Laplace transformation technology was applied with a complementary error basis function to find analytical solutions of stress evolution in multibranch interconnect trees during void nucleation. In [9], based on the accelerated separation of variables (ASOVs), an analytical model was proposed to describe stress evolution with Gaussian elimination (GE), yet it can only deal with stress evolution under constant temperature.

Moreover, deep learning has demonstrated the ability to explore the underlying correlation of massive data in computational modeling of physical systems [18]–[21]. In [22], the trial function is configured for satisfying boundary conditions (BCs) and initial conditions (ICs), and a single-hidden-layer neural network is employed as the nonlinear operation component of the trial function. This way offers a new insight for obtaining approximate solutions to PDEs. In [23], structural similarity between residual neural networks (ResNets) and PDE was observed to establish a new PDE interpretation of convolutional neural networks (CNNs). Research in [24]...
introduced hidden physics models (data-efficient learning machines) to extract patterns from high-dimensional data governed by time-dependent and nonlinear PDEs. In [25], a weak adversarial network (WAN) was proposed for high-dimensional PDEs by leveraging their weak formulations. The primal and adversarial networks are set to minimize the converted objective function. Furthermore, the multifidelity physics-informed neural network (MPINN) provides a composite structure to exploit the relationship between high-fidelity data and low-fidelity data of PDEs to obtain the PDE solutions [26]. Inspired by these works, a physics-constrained deep learning scheme-based solver was proposed for analyzing electric potential and electric fields in [27], which provides new insight for applying deep learning in EM assessment.

In this article, we propose a space–time conversion-based multichannel network motivated by the physics-informed neural network (PINN) [28], which encodes PDEs respecting given laws of physics into the neural networks for learning tasks. The proposed network, called space–time PINN (STPINN), aims at calculating stress evolution in multisegment interconnect trees during the void nucleation phase under complex temperature conditions. It should be noted that the space and time in the proposed STPINN method means that the diffusivities depend on both location and time. The proposed STPINN method is a mesh-free approach that can obtain the continuous and differentiable stress solution without a mesh generation since STPINN applies automatic differentiation (AD) [29] for all the derivative operations to enforce the strong form of the stress evolution equation [30]. Mesh is needed to be created in numerical methods, such as FEM and FDM for the discretization in the space domain. The main contributions of this article are as follows.

1) We propose the STPINN architecture for the first time for solving the mesh-free stress evolution problem in EM-based reliability analysis without requiring a prior knowledge of the relevant datasets.

2) We extend the proposed STPINN method from analyzing EM-based reliability of a single wire to interconnect with multisegments. STPINN can be used to solve the equations describing stress evolution on multisegment interconnect trees under space–time-related diffusivity.

3) We consider the effects of the dynamic temperature incorporating Joule heating and the via effect in the proposed STPINN for calculating the EM-induced stress evolution during the void nucleation phase.

4) We compare the proposed method against the FEM and the compact analytical model. The proposed model is 2×–52× faster than existing methods, with mean relative errors < 1.22% versus FEM and < 0.44% versus the analytical model.

The remainder of this article is organized as follows. Section II describes the physics-based Korhonen’s equation for EM modeling and the thermal model considering Joule heat, via effect and time-varying temperature. In Section III, we introduce the STPINN and the corresponding preprocessing procedure. Experiments are presented in Section IV for the configured multisegment interconnect trees in analyzing stress evolution under different thermal models, followed by the conclusion in Section V.

II. BACKGROUND OF PHYSICS-BASED EM MODELING

The physics-based stress evolution model governed by Korhonen’s equation demonstrates its reliability in stress evolution estimation compared to empirical methods. The space–time-related temperature model shows the temperature distribution through the heat conduction modeling. In this section, we will introduce the general EM modeling and the thermal model used in EM analysis.

A. Stress Evolution Model

EM phenomenon results from a combined action of the momentum exchange with the conducting electrons and the electrostatic force. The electrons and metal atoms collide within the high-density current. During the collision, the metal atoms are driven by the electronic wind force and the opposite mechanical driving force, shown in Fig. 1. As a result, when the metal line is embedded into the rigid confinement, the atoms are subject to tensile stress at the cathode and compressive stress at the anode, causing depletion and accumulation of volume changes, respectively. Void nucleation can be determined when tensile stress exceeds the critical stress, called $\sigma_{\text{crit}}$. It is worth mentioning that void nucleation will first occur at the cathode node due to the atomic migration.

The physics-based model of Korhonen et al. [16] governs hydrostatic stress evolution during the void nucleation phase with PDEs. Let $\sigma(x,t)$ denote the stress evolution at location $x$ and time $t$. The stress evolution along a single metal wire in one dimension takes the following form:

$$
PDE : \frac{\partial \sigma(x,t)}{\partial t} = \frac{\partial}{\partial x} \left[ \kappa \left( \frac{\partial \sigma(x,t)}{\partial x} + G \right) \right], x \in \mathcal{L}
$$

$$
BC : \kappa \left( \frac{\partial \sigma(x,t)}{\partial x} + G \right) = 0, x \in \mathcal{B}
$$

$$
IC : \sigma(x,0) = \sigma_T, x \in \mathcal{L}
$$

where $t$ satisfies $t \in (0, +\infty)$. The notations $\mathcal{L}$ and $\mathcal{B}$ represent the set of points within the metal wire and the collection of points at the blocked terminals. The notations $\kappa = D_0 B \Omega/(kT)$ and $B$ are stress diffusivity and effective bulk related to line geometry, especially width, aspect, and grain morphology [31]. Whereas $k$ and $T$ are Boltzmann’s constant and the absolute temperature, respectively. The effective atomic diffusion coefficient is expressed as follows:

$$
D_a = D_0 \exp \left( -\frac{E_a}{kT} \right).
$$

Here, $D_0$ is the self-diffusion coefficient and $G = |Z|^2 e\rho / \Omega$ is the EM driving force. The atomic lattice volume, current density, metal resistivity, the effective charge number, and activation energy are denoted by $\Omega, j, \rho, Z^*, \text{ and } E_a$, respectively. The notations and their values in this article are summarized in Table I.

In (1), the stress evolution $\sigma(x,t)$ is simultaneously described by Korhonen’s PDEs, zero-flux BCs, and ICs. Here, $\sigma_T$ is the preexisting stress along the wire. In our work, we assume $\sigma_T$ to be zero. During the void nucleation phase, atomic flux is blocked at the boundary of wire geometry in corresponding dimensions and the atomic flux [16] is given by

$$
J(x,t) = \frac{D_a C \Omega}{kT} \left( \frac{\partial \sigma(x,t)}{\partial x} + G \right)
$$

Fig. 1. Atomic forces on single-segment wire due to high-density current.
TABLE I
DESCRIPTION AND TYPICAL VALUE OF PARAMETERS
IN THE EXPERIMENTS

| Parameter | Value               | Description                      |
|-----------|---------------------|----------------------------------|
| $k$       | $1.38 \times 10^{-23}$ J/K | Boltzmann constant              |
| $e$       | $1.6 \times 10^{-19}$ C   | Electric charge                  |
| $Z^*$     | 10                  | Effective valence charge         |
| $E_a$     | 1.1eV              | Activation energy                |
| $B$       | $1 \times 10^{12}$ Pa    | Effective bulk                   |
| $D_0$     | $5.2 \times 10^{-5}$ m$^2$/s | Self-diffusion coefficient    |
| $\rho$    | $3 \times 10^{-10}$ g/m$^3$ | Resistivity of Cu               |
| $\Omega$  | $8.78 \times 10^{-24}$ m$^3$ | Atomic volume                   |
| $\sigma_{\text{crit}}$ | $4 \times 10^8$ Pa    | Critical stress                  |

where $C$ represents the number of metal atoms per unit volume. In multisegment topology, the value of stress and atomic flux should be continuous at the interior junction nodes of adjacent segments. For an interconnect tree with multisegments, Korhonen’s equations can be extended to the following form to describe the stress evolution $\sigma_i(x, t)$ along the $i$th segment:

PDE: \[
\frac{\partial \sigma_i(x, t)}{\partial t} = \frac{\partial}{\partial x} \left[ k_i \left( \frac{\partial \sigma_i(x, t)}{\partial x} + G_i \right) \right], \quad x \in L_i
\]

BC: \[\kappa_i \left( \frac{\partial \sigma_i(x, t)}{\partial x} + G_i \right) = 0, \quad x \in B\]

IC: \[\sigma_i(x, 0) = 0, \quad x \in L_i.\]

We employ $I$ to describe the collection of coordinates of interior junction nodes. The collection of adjacent segments of each interior junction node on the $i$th segment is defined as $N_i = \{i_1, \ldots, i_p\}$. The notation $n_j$ represents the unit inward normal direction of the interior junction node on the neighboring $j$th segment, of which the value (1 or $-1$) is determined by the direction of electron propagation [17]. The BCs at both interior junction nodes of each segment should be satisfied. It is remarked that the impacts of constant, time-varying, and space–time-dependent temperatures are quantified with $\kappa_i$, leading to three different types of PDEs for the stress evolution model.

B. Space–Time-Related Temperature Model

In EM reliability analysis, the temperature is one critical factor affecting the EM failure process. The temperature variation was taken into account by [32]–[34] in EM analysis. In Fig. 2, the simulation [8] reveals that the temperature rising will lead to an early failure of the full chip. The temperature distribution of the interconnect trees depends on the gate switching activity and the quality of heat dissipation. The total thermal power dissipated can be represented as a sum of the average power dissipated by Joule heating of the metal branches and the average heating dissipated by the underlying logic due to active switching and leakage currents [32].

In this section, we review the Joule heat model [35] and combine the model with time-varying temperature to customize the space–time-related temperature model for EM analysis. In the metal wire with a high-density current, the nonuniform temperature rising is generated due to the Joule heat effect. It should be noted that during Joule heat conduction, the estimated temperature will be higher than the practical temperature if the heat dissipation paths are ignored. The impact of low-$k$ dielectrics such as SiO$_2$ on the interconnect temperature was reported in [36]. In copper processes, via serve as more efficient heat dissipation paths since they have much better thermal conductivity than the low-$k$ dielectrics [37]. Thus, the efficient heat dissipation of vias, called via effect, should be considered in temperature estimation during Joule heat conduction. Given the length of metal wire, the space–time-related temperature distribution considering via effect follows [35] as:

\[
T(x, t) = T_0(t) + \frac{j_{\text{ma}}^2 L_H}{k_M k_{\text{ILD}}^2} \left[ 1 - \frac{\cosh \left( \frac{x}{L_H} \right)}{\cosh \left( \frac{L}{2} \right)} \right], \quad x \in \left[ -\frac{L}{2}, \frac{L}{2} \right]
\]

(5)

where $L$, $\rho$, $k_M$, and $j_{\text{ma}}$ are the length, the resistivity, the thermal conductivity, and the uniform root-mean-square current density of the metal wire, respectively. The time-varying model $T_0(t)$ represents the temperature on the underlying layer and characterizes the impacts of the gate switching activity and the quality of heat dissipation. The notation $L_H$ denotes the healing length [35] and it can be further explained in detail as follows:

\[
L_H = \left[ \frac{k_M H_{\text{ILD}}}{k_{\text{ILD}}} \left( \frac{1}{s} \right) \right]^{\frac{1}{2}}
\]

(6)

Here, $s$ is the heat spreading factor, and the notations $d$ and $w$ are the spacing and width of the metal lines. The notations $H_{\text{ILD}}$ and $k_{\text{ILD}}$ are the thickness and the conductivity of interlayer dielectric (ILD), which separates the metal wire and underlying layer. The temperature on the multisegment interconnect tree is assumed to be a combination of the nonuniform distributed temperatures on each segment calculated by (5). For the node connecting to more than one segment, we suppose that there is negligible heat loss between the node and the underlying layer [35]. Based on (5), the temperature of the node over time can be expressed as follows:

\[
T\left( \pm \frac{L}{2}, t \right) = T_0(t)
\]

(7)

In this work, we consider three temperature scenarios: I) constant temperature over space and time; II) time-varying temperature; and III) space–time-related temperature considering Joule heat spreading, via effect and time-varying
III. STPINN-BASED EM RELIABILITY ANALYSIS

In this section, we first modify the PINN structure in EM analysis under constant thermal condition and then extrapolate the PINN method to a novel STPINN-based EM reliability analysis model for accurate mesh-free stress evolution on multisegment interconnects during the void nucleation phase under different temperature conditions. Fig. 3 shows an overview flowchart of the learning-based EM model which consists of data preparation, model training, and stress solution. The training data are sampled from the configured geometry and time range and then preprocessed to regular parameters for learning. During the training phase, the network weights are continually updated until the objective function converges to the minimum. The learned model first obtains normalized stress by the needed location and time (test data) and then restores the normalized stress through scaling to achieve the EM stress evolution along the configured interconnect tree. More details in the flowchart will be discussed.

A. Modified Physics-Informed Neural Network

PINNs have demonstrated the strong expressive powers in solving forward and inverse PDE problems through their nonlinear learning capabilities. Application in learning velocity and pressure fields by partially observing space–time visualizations of a passive scalar [38] relies on both data and physics-based information described by PDEs [39]. We now enhance and customize the PINN method for solving the interconnect reliability problem.

To fit into EM-based reliability problem, we first transform Korhonen’s stress evolution equation (1) to the following form:

\[ \sigma_t + \mathcal{N}[\sigma] = 0 \]  

where \( \mathcal{N}[\sigma] = \partial(\kappa(\partial\sigma/\partial x + G))/\partial x \) and \( \sigma_t = \partial\sigma/\partial t \) represent the nonlinear differential operator and the first derivative of \( \sigma \) with respect to temporal variable \( t \), respectively. We also define the diffusion operator \( f \) which is employed to substitute the left side of (8) and \( f \) is expected to be zero.

It has been shown in [40] that a feedforward neural network (FNN) is a useful tool for function approximation with the practical benefits of modeling highly nonlinear functions. The structure of FNN is intuitive and straightforward to understand and manipulate. Thus, we use the FNN to approximate the solution \( \sigma \) and take spatial–temporal information \((x, t)\) as inputs. It is worth mentioning that the EM driving force is not the same in different segments due to various current densities within each segment. According to the BCs (4) corresponding to the interior junction node constraints, derivatives of stress evolution with respect to the location at the interior junction node in different directions are different. Therefore, the stress solution \( \sigma \) of the whole interconnect tree is nondifferentiable at the interior junction nodes and is hard to be approximated by the multiorder differentiable activation function. In order to overcome this problem, we insert a virtual distance \( \nu \) at the intersection of adjacent segments and set the BCs corresponding to the interior junction node constraints at the terminals of the virtual distance, which are the overlapping points at the intersection node of the interconnect structure, shown in Fig. 4. It should be noted that derivatives of the multiorder differentiable output of PINN in different directions with respect to the location need to be equal at any time. However, this is incompatible with the interior junction node constraints so that we cannot directly construct the objective function without introducing a virtual distance at the interior junction node. The virtual distance smooths the stress evolution and helps

![Fig. 3. Flowchart of the learning-based EM reliability analysis model.](image-url)

![Fig. 4. Schematic of virtual distance for calculating the derivatives of stress evolution with respect to the location at the interior junction node. The blue nodes (right) are overlapping at the interior junction (left).](image-url)
to construct the objective function for network training in our proposed STPINN method. The virtual distance is set to be \( v = 0.5 \mu m \) in our experiment and the objective function is finally defined as follows:

\[
\text{MSE} = \text{MSE}_f + \text{MSE}_b + \text{MSE}_i + \text{MSE}_c. \tag{9}
\]

In (9), \( \text{MSE}_f \) represents the mean-square error of diffusion operator \( f \) and describes the relationship between the nonlinear operator term \( N[\sigma] \) and the temporal term \( \sigma_t \). The losses \( \text{MSE}_b, \text{MSE}_c, \) and \( \text{MSE}_i \) correspond to BCs at terminals, BCs at interior junction nodes, and ICs, respectively. When solving the stress using the modified PINN and the proposed STPINN, each term on the right side of (9) is defined as follows:

\[
\text{MSE}_f = \frac{1}{N_f} \sum_{l=1}^{N_f} \left[ \frac{\partial \sigma(x_l, t_l)}{\partial t} - \frac{\partial}{\partial x} \left( \kappa(x_l, t_l) \left( \frac{\partial \sigma(x_l, t_l)}{\partial x} + G_l \right) \right) \right]^2
\]

\[
\text{MSE}_b = \frac{1}{N_b} \sum_{l=1}^{N_b} \kappa(x_l, t_l) \left( \frac{\partial \sigma(x_l, t_l)}{\partial x} + G_l \right)^2, \quad x \in B'
\]

\[
\text{MSE}_i = \frac{1}{N_i} \sum_{l=1}^{N_i} \left[ \sigma(x_l, 0) - \sigma(x_l, t_l) \right]^2
\]

\[
\text{MSE}_c = \frac{1}{N_c} \sum_{l=1}^{N_c} \sum_{m=1}^q \left( \sum_{x \in \Omega_m} |\sigma(x, t_l) - \sigma(c_m, t_l)| \right)^2 + \sum_{x \in \Omega_m} \kappa(x, t_l) \left( \frac{\partial \sigma(x, t_l)}{\partial x} + G_l \right) \cdot n_j \left( \sum_{x \in \Omega_m} \kappa(x, t_l) \left( \frac{\partial \sigma(x, t_l)}{\partial x} + G_l \right) \cdot n_j \right)^2 \tag{10}
\]

where \( x_l \in L' \) in \( \text{MSE}_f \). Notations \( L' \) and \( B' \) are the set of points within all segments and at blocked terminals, respectively, extracted from interconnect trees inserted with virtual distances at interior junction nodes. It is supposed that there are \( q \) interior junction nodes in the interconnect tree. We denote the collection of overlapping nodes describing the \( m \)th interior junction node located at \( c_m \) by \( \Omega_m = \{c_m, \Omega_m\} \) where \( \Omega_m = \{c_m, 1, \ldots, m\} \) represents the collection of coordinates of the neighboring nodes at virtual distances connected with the \( m \)th interior junction node. The notations \( G_i, G_b, \) and \( G_j \) are the EM driving forces on the segment corresponding to \( x_l \), the blocked segments, and the segment connected with node \( x_l \), respectively. The notation \( t_l \) denotes the time point sampled randomly within the configured time range. For the PINN analysis, we first obtain the temperature \( T(x, t) \) according to the thermal model (5) and then calculate the diffusivity \( \kappa(x, t) = D_s B / (kT(x, t)) \) through the obtained temperature at a certain time and location. The mean-square error \( \text{MSE}_f \) can be calculated by (10) through collocation points randomly sampled in space and time domains and the corresponding diffusivity \( \kappa(x, t) \). After that, we can calculate the objective function \( \text{MSE} \) by the losses \( \text{MSE}_f, \text{MSE}_b, \text{MSE}_c, \) and \( \text{MSE}_i \), which is used for the PINN training.

The PINN model works as a solver for predicting the stress evolution with differently configured currents flowing in multiple interconnected segments. Compared with general supervised neural network methods, training data preparation of PINN omits data labeling since the objective function is only related to the sampled location and time point. However, one potential limitation of the PINN-based model for calculating EM-based stress evolution stems from the learning ability and approximate capacity of the network structure for solving PDE with different diffusivities. It is required to improve the network structure to cope with different diffusivities such that we can obtain an exact stress solution under complex temperature conditions.

B. Gradient Computation

Before employing the neural network-based method for stress assessment, we first analyze the gradient of neural networks during the training phase. We use an FNN consisting of two input units, one hidden layer with \( H \) tanh units, and one linear output unit to approximate the stress evolution. The input vector is defined as \( a_j (j = 1, 2) \) comprising of the coordinate of the location and the time instance. The output is \( N = \sum_{j=1}^H v_j o(z_j) \) where \( z_j = \sum_{i=1}^2 w_{ij} a_j + u_i \). Here, \( u_i \) and \( w_{ij} \) denote the bias of hidden unit \( i \) and the weight from the input unit \( j \) to the hidden layer unit \( i \). The notation \( v_i \) denotes the weight from the hidden unit \( i \) to the output and \( o(z) \) is the tanh activation function. Then, the derivative term of the stress evolution in the objective function follows:

\[
\frac{\partial^k N}{\partial a_{ij}^k} = \sum_{i=1}^H v_j w_{ij} o^{(k)}(i) \tag{11}
\]

where \( o_i = o(z_i) \) and \( o^{(k)} \) is the \( k \)th-order derivative of the tanh function. Thus, the gradient of each derivative term in (11) can be obtained as follows:

\[
\frac{\partial^k N}{\partial a_{ij}^k} = w_{ij} o^{(k)}(i) + v_j w_{ij} o^{(k+1)}(i) + u_i v_j w_{ij} o^{(k+1)}(i) + v_j k w_{ij} o^{(k)}(i). \tag{12}
\]

For the parameters in the objective function, the magnitudes of \( x, t \), and the EM driving force are close to \( 1 \times 10^{-7}, 1 \times 10^8 \), and \( 1 \times 10^{15} \), and the value of \( \kappa \) in Korhonen’s function is about \( 1.413 \times 10^{-18} \) under the temperature \( T = 350 \) K. The gradient of the objective function with respect to the network parameters can be defined based on (9), (10), and (12). Fig. 5(a) shows the absolute gradient value \( |\text{Gradient}| \) along with the training iterations. It can be observed that the maximum value of \( |\text{Gradient}| \) is about \( 3.1 \times 10^{15} \) and the value of \( |\text{Gradient}| \) after convergence is about \( 2.9 \times 10^8 \) with minor fluctuations. The
learned model cannot provide accurate stress solutions through the large gradient. The constraints in Korhonen’s equation will not be satisfied if we directly normalize the parameters. To deal with the problem, we develop a linear preprocessing method for the EM model.

C. Preprocessing and Data Preparation

In this section, we propose a preprocessing method for EM analysis to handle the gradient problem. We first unfold the PDE, BC, and IC in (1) to Euler format and perform linear transformation

\[
PDE : \frac{\hat{\sigma}^{m+1} - \hat{\sigma}^m}{\tau} = \frac{\omega^2}{\omega_k} \left( \frac{\hat{\sigma}^{m}_{j-1} - 2\hat{\sigma}^m + \hat{\sigma}^m_{j+1}}{h^2} \right) = 0
\]

\[m = 0, 1, \ldots, M, j = 1, 2, \ldots, N - 1\]

\[
BC : \alpha m - \alpha_0^m = -\hat{G}_1, \quad \hat{\sigma}^m - \hat{\sigma}_N^m - \hat{\sigma}_{N-1}^m = -\hat{G}_2
\]

\[\text{IC : } \hat{\sigma}^0 = \omega_0 \sigma(x, 0). (13)\]

We denote \(t\) and \(h\) as the time and length interval for time and space discretization and construct numerical relationship \(\hat{\sigma} = \omega_0 \sigma, \hat{h} = \omega_0 h, \hat{\tau} = \omega_0 \tau, \text{ and } \hat{G}_i = \omega_0 / \omega_k G_i (i = 1, 2)\) for the linear-scale transformations on \(h, \tau, \sigma, \text{ and } G_i\). Here, \(\omega_0, \omega_k, \text{ and } \omega_m\) represent scaling factors of the length, the time, and the stress \(\sigma\), respectively. Number of mesh grids in the space domain \(N\) is related to the transformed length interval \(\hat{h}\) with the relationship \(N\hat{h} = \omega_0 L\), where \(L\) is the total length of wire. The temporal domain is configured to \(t \in [0, T]\). The notation \(T_s\), selected for simulating the reliability assessment time range, satisfies the relationship \(M\hat{t} = \omega_0 T_s\). In addition, the scaling factor of the stress \(\omega_0\) is unrelated to PDEs and BCs for its elimination in both sides of the formulas. Let \(\sigma(x, t, \kappa)\) and \(\hat{\sigma}(x, t, \kappa)\) be the solutions of original and normalized expressions. Then, the solution of stress evolution equation satisfies

\[\sigma(x, t, \kappa) = \omega_0 \hat{\sigma}(\omega_x x, \omega_t t, \omega_k^2 \omega_4 \kappa). (14)\]

The inputs of networks are converted to the regular parameters consisting of variables \(\omega_x x\) and \(\omega_t t\). Correspondingly, the diffusion coefficient in both the stress evolution equation and the objective function is rewritten as \(\omega_k^2 / \omega_4 \kappa\). Fig. 5(b) shows the absolute value of the gradient after preprocessing along with training iterations. It can be observed that \(|\text{Gradient}|\) fluctuates between \(5.5 \times 10^{-4}\) and \(4.7 \times 10^{-1}\). The learned model can achieve satisfactory accuracy after employing the preprocessing scheme. This procedure plays an important role in the application of the learning-based method for solving stress evolution equation in EM reliability analysis.

For the original data preparation of preprocessing on solving PDEs, Latin hypercube sampling (LHS) [41] is applied to generate spatial–temporal input data in the PINN method [28]. Based on the observation of the transformed diffusion coefficient and the time range in the stress evolution equation for VLSI reliability analysis, logarithmic isometric sampling (LIS) demonstrates its rationality in sampling since the gradient is relatively steep near the zero point. In test cases on the neural network-based model, we sample the spatial location by LHS and the time point by both LIS and LHS and then generate the training data and test data by preprocessing the collection of spatial location and time points, shown in the data preparation procedure of Fig. 3.

D. STPINN

It is reported that the PINN method can be used to solve some types of PDEs (such as constant diffusivity) [28]. However, the accuracy of the PINN method depends on the complexity of the diffusivities. The PINN method cannot well capture the stress evolution process when the temperature varies with respect to space and time. To solve the problem, we propose the STPINN by developing new space and time variables for PDEs with temperature-dependent diffusivity. We suppose the stress \(\sigma(x, t)\) can be divided to \(n\) solutions of different diffusion process on various spatial–temporal space, which can be expressed as follows:

\[\sigma(x, t) \approx \sum_{k=0}^{n} \omega(x_k, t_k, E_k(x_k, t_k)) (15)\]

where \(\omega(x_k, t_k, E_k(x_k, t_k))\) is the solution of the diffusion process with location \(x_k, \text{ time } t_k, \text{ and diffusivity } E_k(x_k, t_k)\). The stress evolution satisfies \(\sigma(x, t) = u(x, t, \kappa(x, t))\). It should be noted that \(x_k, t_k, \text{ and } E_k(x_k, t_k)\) are new spatial, temporal transformed variables, and the unknown basis diffusion coefficient in the \(k\)th component of the stress, respectively. We define \(\omega(x_k, t_k, E_k(x_k, t_k)) = u_k\) and rewrite the diffusion equation of \(u_k\) as follows:

\[\frac{\partial u_k}{\partial t} = \frac{\partial}{\partial x} \left[ E_k(x_k, t_k) \left( \frac{\partial u_k}{\partial x} + G \right) \right]. (16)\]

We define \(\partial t / \partial t_k = \gamma_k(t)\) and \(\partial x / \partial x_k = \xi_k(x)\) and introduce a new temporal variable \(t_k\) which satisfies the following equation:

\[\frac{\partial t_k}{\partial t} = \frac{\xi_k^2(x)}{E_k(x_k, t_k)} (\gamma_k). (17)\]

Moreover, we define the notation \(u_k = u(x, t_k, E_k(x_k, t_k))\) which follows the diffusion equation:

\[\frac{\partial u_k}{\partial t_k} = \frac{\partial}{\partial x} \left[ E_k(x_k, t_k) \left( \frac{\partial u_k}{\partial x} + G \right) \right]. (18)\]

According to (16)–(18), the \(k\)th component \(u_k\) can be calculated as follows:

\[u_k = \xi_k(x) u_k. (19)\]

In (18), we define \(E_k(x_k, t_k) = E(x, t_k)\), where \(E(x, t_k)\) is an unknown basis diffusion coefficient. In this way, the shared parameters can be employed to obtain \(u_k\) through the inputs \(x\) and \(t_k\). It can be observed in (17) that the transformation from \(t\) to \(t_k\) is related to the transformation from \(x\) to \(\hat{x}\). For simplify, we suppose that a linear transformation is executed on \(x\), which satisfies \(\xi_k(x) = \theta_k\). Thus, we can derive the stress evolution solution \(\sigma(x, t)\) in (1) as follows:

\[\sigma(x, t) \approx \sum_{k=0}^{n} \omega(x_k, t_k, E(x, t_k)) (1)\]

\[t_k = \int_0^t \theta_k(x')/\gamma_k(t') dt'. (20)\]

The proposed STPINN method is motivated by the spectral method in the mathematical intuition, which expands the solution approximately into a finite series expansion of a smooth function. In this way, it is supposed that the solution of the stress evolution equation can be divided into several
smooth solutions respecting different subdiffusion processes in different spatial–temporal spaces.

Based on (20), we build a network consisting of neural network-based linear and nonlinear operators for variable transformation in the proposed STPINN architecture. The proposed method comprises three connected neural networks with multichannels, as shown in Fig. 6. The first network \( \tilde{\mathcal{F}}_f(t; \alpha) \) is employed as the nonlinear temporal transformation operator in (17), which converts a time variable \( t \) into \( n \) new sub variables \( t_k (k = 1, 2, \ldots, n) \). The second FNN \( \mathcal{F}(x, t; \beta) \) obtains the mesh-free solutions of PDEs with the diffusivity \( E(x, t_k) \). Several couples of outputs \( \tilde{\mathcal{F}}_c(k = 1, 2, \ldots, n) \) perceived by \( \mathcal{F}(x, t; \beta) \) are connected to the fully connected layer \( \mathcal{F}_c(\tilde{\mathcal{F}}; \theta) \), which operates spatial conversion in (19). The original data are sampled through the method mentioned in Section III-C into \( \mathcal{G}_f, \mathcal{G}_i, \mathcal{G}_b, \) and \( \mathcal{G}_c \) and then they are preprocessed into \( \tilde{\mathcal{G}}_f, \tilde{\mathcal{G}}_i, \tilde{\mathcal{G}}_b, \) and \( \tilde{\mathcal{G}}_c \), which are collocation points for the diffusion operator, ICs, BCs at terminals, and BCs at interior junction nodes.

For the number of STPINN channels \( n \), we choose different \( n \) according to different diffusion coefficient types. For instance, a one-channel STPINN (1-STPINN) has sufficient capacity for solving the PDE with time-dependent diffusivity \( \kappa(t) \) in case II, where no spatial-dependent transformation is caused by the nonconstant diffusivity. We multiply both sides of the PDE in (1) by \( \partial_t \sigma / \partial T' \), where \( T' = \int_0^t (\kappa(t') / \kappa_0) dt' \). The transformation follows:

\[
\frac{\partial \sigma}{\partial t} \frac{\partial t}{\partial T'} = \frac{\partial \sigma}{\partial x} \left[ \kappa(t) \frac{\partial t}{\partial T'} \left( \frac{\partial \sigma}{\partial x} + G \right) \right], \quad x \in \mathcal{L}. \tag{21}
\]

Then, the PDE with new temporal variable takes the following form:

\[
\frac{\partial \sigma}{\partial T'} = \frac{\partial \sigma}{\partial x} \left[ \kappa_0 \left( \frac{\partial \sigma}{\partial x} + G \right) \right], \quad x \in \mathcal{L}. \tag{22}
\]

Here, the time-related diffusivity PDE can be converted to a new one with constant diffusivity \( \kappa_0 \) through the transformation on temporal variable \( T' \). Fig. 7 shows the comparison between the results of the transformed variable \( T' \) obtained by the neural network-based solver and the numerical Runge–Kutta method. It can be seen from Fig. 7 that the results by the neural network method fit well with the numerical method within 0.2% relative error. The neural network-based solver consumes 0.0003 s for calculating \( T' \) and the Runge–Kutta method costs 0.002 s for calculating \( T' \) in 100-time steps.

The objective function is imposed by (9) and (10) through the training datasets \( \tilde{\mathcal{G}}_f, \tilde{\mathcal{G}}_i, \tilde{\mathcal{G}}_b, \) and \( \tilde{\mathcal{G}}_c \) as shown in Fig. 6. The outputs of the network are restored based on stress scaling in preprocessing. Hence, STPINN enhances the ability of the neural network to learn approximate solution of complex PDEs. It should be noted that the structure of STPINN is not just adding neurons and hidden layers over PINN. The proposed STPINN method establishes a constraint for approximating each subdiffusion process and combining the solutions together to achieve an accurate approximation of the stress evolution under complex diffusivities. The proposed STPINN method with multichannels employs the same MLP to solve each subdiffusion process, which decreases the demand of the number of trainable weights.

In order to employ STPINN for EM stochasticity assessment, an atomic diffusivity \( D_0 \) can be obtained for the interconnect tree in each Monte Carlo iteration. For case I of constant temperature analysis, the stress evolution \( \sigma^*(x, t) \) with \( D_0 \) can be obtained by

\[
\sigma^*(x, t) = \sigma(x, \frac{D_0}{D_0} t) \tag{23}
\]
which can be derived by
\[
\frac{\partial \sigma^*(x, t)}{\partial t} = \frac{\partial}{\partial x} \left[ \frac{D_a}{D_0} \frac{\partial \sigma^*(x, t)}{\partial x} \right] + \frac{\partial}{\partial x} \left( \frac{D_0}{D_a} \frac{\partial \sigma^*(x, t)}{\partial x} + G \right). \tag{24}
\]

In (23), \( D_a \) is a determined atomic diffusivity at a constant temperature \( T \) used for neural network training and \( \sigma(x, t) \) is the stress solution under \( D_a \). For case II of time-varying temperature, the stress evolution \( \sigma^*(x, t) \) with \( D_a^* \) can be obtained through 1-STPINN as follows:
\[
\sigma^*(x, t) = F(x, \frac{D_a^*}{D_a} F_t(t; \alpha); \beta). \tag{25}
\]

In (25), \( F(x, t; \alpha) \) is trained for the stress evolution analysis with the determined atomic diffusivity \( D_a \). In this way, the proposed STPINN may capture the randomness in EM degradation by employing the trained STPINN model several times under time-varying temperature.

In the next section, experiments will demonstrate the effectiveness of the STPINN structure in solving stress evolution equation under complex temperature.

IV. EXPERIMENTAL RESULTS

A. Experiment Setup

We demonstrate the thermal effects on interconnect wires with three cases stated in Section II-B. The proposed model is compared with the FEM [43] and the compact analytical method [44]. Experimental results will show that the proposed STPINN method can be also applicable to interconnect wires with multisegments. The proposed STPINN method is implemented in Python 3.6.2 with Tensorflow 1.12.0, and the compact analytical method is implemented in Python 3.6.2. The experiments are carried out on a 3.0-GHz PC with 8-GB of RAM.

In the training process, we first perform Adam optimization [45] for 5K iterations and then use L-BFGS [46] for the subsequent training. We remark that the second-order derivative-based L-BFGS can achieve better accuracy with less iterations than the first-order derivative-based Adam [30]. Our experiments show that the combination of Adam and L-BFGS can achieve better training performance, shown in Fig. 8. During the training phase, we use the Adam algorithm to calculate the initial weights of L-BFGS for avoiding stacking at a bad local minimum. The initial learning rate of Adam is set to be 0.001 together with Xavier’s initialization method, and the numbers of training data are set to be \( N_f = 25000, N_b = N_c = 1000, \) and \( N_0 = 500, \) respectively. The scaling factors \( \omega_{\alpha}, \omega_{\beta}, \) and \( \omega_{\gamma} \) are set to be \( 1 \times 10^{-9}, 1 \times 10^3, \) and \( 1 \times 10^{-7}, \) respectively. Since there are multiorder differential operations in the objective and optimization functions, a multiorder differentiable activation function is required for the nonlinear transformation in the proposed STPINN method. In this work, we choose tanh as the activation function since the experimental results show that the tanh function has a better approximating ability in the EM model than the sigmoid function. Geometric and current density configuration of the interconnect wire with two segments is set as follows:
\[
\mathcal{L}_1 : \{x : 0 < x < 20 \ \mu m\}, \quad j_1 = 4 \times 10^{10} \ \text{A/m}^2,
\]
\[
\mathcal{L}_2 : \{x : 20 \ \mu m < x < 50 \ \mu m\}, \quad j_2 = -1 \times 10^{10} \ \text{A/m}^2.
\tag{26}
\]

For the comparisons of FEM, PINN, and STPINN, the typical values of the parameters used for calculating the stress evolution of this two-segment structure are shown in Table I. The original self-diffusion coefficient \( D_0 \) is to keep the same in this two-segment structure. The effective atomic diffusion coefficient \( D_a \) is calculated by (2).

B. Experimental Results

In case I, the constant temperature is set to be 350K in FEM, PINN, and STPINN. Based on Korhonen’s equation, we use a PINN consisting of ten hidden layers with 40 neurons per layer for obtaining the stress evolution under constant temperature. It can be seen in Fig. 9 that the approximate stress solutions along with the segments at different times obtained by PINN agree well with the numerical results obtained by the FEM. Fig. 10 shows the absolute error defined by \( |\sigma_{\text{PINN}} - \sigma| \), illustrating that the absolute error is mainly in the range \(-0.0015\)–\(-0.0005\) GPa and the amplitude of error grows near the terminals with the increase of time. The distribution of error fits well with the trend of hydrostatic stress evolution. The results demonstrate the promising capability of PINN in solving stress evolution under constant temperature. However, PINN cannot well capture the EM-induced stress evolution under space–time-related temperature, which will be discussed in Section IV-C.

In case II, the dynamic temperature is configured as \( T = (35 + 30 \times \sin(4 \times 10^{-8} \pi t))K \), which is used in the simulations of both FEM and STPINN. We employ a one-channel
Fig. 10. Absolute error of EM stress development under case I between the proposed model and FEM along two-segment interconnect tree in the time range 0–10^8 s.

Fig. 11. Comparisons of EM stress development under case II between the proposed model and FEM along two-segment interconnect tree at different time.

Fig. 12. Absolute error of EM stress development under case II between the proposed model and FEM along two-segment interconnect tree in the time range 0–10^8 s.

STPINN (1-STPINN) consisting of one hidden layer with 100 neurons per layer in \( \mathcal{F}_t(t; \alpha) \) and ten hidden layers with 40 neurons per layer in \( \mathcal{F}(x; t; \beta) \) for this case. The first FNN \( \mathcal{F}_t \) represents the integration operation of temporal variable \( t \) for converting the time-dependent diffusivity toward constant diffusivity in the stress evolution equation. The comparison results between the FEM and the proposed STPINN method are shown in Figs. 11 and 12. The relative error is less than 1.22% and the absolute error is mainly oscillating from \(-0.002\) to \(0.008\) GPa. The relatively large error is distributed near the terminals of the interconnect wire.

In case III, for the comparisons of FEM and STPINN, the space–time-related temperature is configured as the thermal model (5) where we set \( H = H_{LD} = 0.8 \) μm, \( w = d = 0.3 \) μm, and \( T_0(t) = [350 + 30 \times \sin(4 \times 10^{-8} \pi t)] K \). The temperature profile is shown in Fig. 13. We assume low-\( k \) insulator SiO₂ (\( k_{oxide} = 1.2 \) W/mK) as ILD, which is more deeply affected by via effect. For this case, we use the two-channel STPINN (2-STPINN) consisting of two hidden layers with 50 neurons per layer in \( \mathcal{F}_t(t; \alpha) \) and ten hidden layers with 40 neurons per layer in \( \mathcal{F}(x; t; \beta) \). The stress evolution profile is shown in Fig. 14, illustrating good agreement with less than 0.80% error at \( t = 5 \times 10^5 \) s, \( 5 \times 10^6 \) s, and \( 5 \times 10^7 \) s. The absolute error distribution shown in Fig. 15 indicates a relatively sharp and large error fluctuation at the junctions near the initial time. Due to the complex diffusivity, the accuracy of the numerical solution decreases when configuring regular mesh grids and time steps in the FEM.

In order to illustrate the stress evolution under different temperature conditions, we compare the transient stress profiles at each junction under cases I–III shown in Fig. 16(a) and (b), which also demonstrate good agreements between the proposed model and the FEM. It can be seen that when taking the thermal conditions with the same mean value into consideration, the stress of cases II and III reach the steady state earlier than the stress of case I. As shown in Fig. 16(a), the stress under case III always maintains a higher value than the
TABLE II

| Case | STPINN | FEM | Chen et al.(2016) | Speed-up |
|------|--------|-----|-----------------|---------|
|      | Level  | Error (%) | Time (s) | Iteration | Error (%) | Time (s) | Speed-up |
|      |        | W.S.    | J.S.    |            | W.S.    | J.S.    | Chen     | FEM      |
| II   | 1      | 1.2220  | 0.1703  | 0.3227     | 100     | 2.0749  | 1.1129   | 0.9595   | 2.97×    | 15.59×   |
|      |        | 5       |         |            | 1000    | 0.9345  | 0.4943   | 1.0082   | 3.12×    | 27.84×   |
| III  | 2      | 0.7976  | 0.1751  | 0.5027     | -       | -       | -        | -        | -       |

W.S. represents whole stress at ten specified time points and J.S. represents junction stress.

Fig. 15. Absolute Error of EM stress development under case III between the proposed model and FEM along two-segment interconnect tree in the time range 0–10⁸ s.

Fig. 16. Comparisons of EM stress development under cases I–III between the proposed model and FEM in time range from 0 to 1 × 10⁸ s at: (a) left terminal and (b) right terminal.

stress under case II at the left terminal. Fig. 16(b) shows that at the right terminal of the interconnect wire, the stress under cases II and III remains almost the same growth rate near the initial time and then decreases in a similar trend, while the stress under case III commences falling earlier. The consideration of Joule heating provides a more effective estimation in EM-based reliability analysis during the void nucleation phase. The higher error in case II is mainly due to the impact of time step selection in the FEM, which will be discussed in Section IV-C.

The comparisons of the proposed STPINN method, FEM, and the compact analytical method [44] are shown in Table II in terms of the accuracy and the computation time. The error of STPINN represents the mean relative error of solutions between STPINN and FEM performing 1K iterations. To further demonstrate the effectiveness of the proposed STPINN, the solutions achieved from STPINN are compared against the results obtained by the compact analytical method under different numbers of iterations. The junction stress accuracy represents the error at the junction of the interconnect tree in the time range 0–10⁸ s and the whole stress accuracy presents the error at ten specified time instances along the wire in the time range 10⁵–10⁸ s. It should be noted that the compact analytical method mainly focuses on the EM-based stress analysis for case II. When calculating the whole stress in the compact analytical method, the mean relative error reduces with increased iterations at the cost of computation time. It can be seen from Table II that the proposed STPINN method can achieve a speedup of 2×–52× over the existing methods with the mean relative error of about 1.2%.

C. Performance Analysis

For analyzing the performance of the proposed STPINN method, Tables III and IV show the following systematic studies of the network configuration, demonstrating the relative errors in the L₂ norm between the STPINN method and the FEM when the architectures of STPINN (F(x, t; β) and F_I(t; α)) are set with different layers and neurons under case III. As shown in Table III, for the network F(x, t; β) with few hidden layers, the increase of neurons per hidden layer will neither reduce the relative error nor accelerate the convergence. When the number of layers rises to 9 and 11, the relative error shows its downtrend with increased neurons. On the contrary, experimental results show that raising the number of layers further will not reduce the relative error nor accelerate the convergence. When the number of layers rises to 9 and 11, the relative error shows its downtrend with increased neurons. On the contrary, experimental results show that raising the number of layers further will not achieve better performance when the number of layers is larger than two. As a result, we can fix the network architecture to two hidden layers with 50 neurons per layer in F_I(t; α) and 11 hidden layers with 40 neurons per layer in F(x, t; β) for stress analysis in the two-segment wire. It can be observed from Tables III and IV...
that changing the structure of $F(x, t; \beta)$ has a greater impact on the accuracy compared with the adjusting network structure of $F(t; \alpha)$. The width and depth of the proposed STPINN architecture are problem dependent on various wire structures. For multisegment wires, we mainly focus on adjusting the structure of $F(x, t; \beta)$. We prefer to first change the width of $F(x, t; \beta)$ and then the depth when adjusting $F(x, t; \beta)$ in the experiments.

In Table V, we report the relative $L_2$ error of STPINN under different number of initial collocation points $N_0$, boundary collocation points $N_b$ and $N_c$, and internal collocation points $N_f$. Here, we set $N_b = N_c = N_0$. It is the general trend that the error is decreased as $N_f$ is increased, given sufficient training points for initial and BCs. For various multisegment wires, we mainly focus on adjusting the number of internal collocation points $N_f$ according to the segment structure as well as the space–time-dependent stress diffusivity for achieving satisfactory loss. During the training phase, the cross-validation method can be employed to monitor whether the network is overfitting.

We further evaluate the impact of the number of channels in the proposed STPINN method for calculating the EM-based stress evolution. We also compare STPINN against PINN in terms of accuracy and computation time. The PINN method is configured in the 10-layer network structure with 40 neurons per layer, and $F(x, t; \beta)$ in the proposed STPINN method is configured as an 8-layer network with 40 neurons per layer. The time mapping network $F(t; \alpha)$ keeps the 2-layer architecture with 40 neurons per layer in the two-channel STPINN and 1-layer architecture with 80 neurons in the one-channel STPINN. Both two methods employ the same optimization technique in the training phase. Table VI shows the accuracy and the runtime of calculating the stress evolution by PINN, one-channel STPINN, and two-channel STPINN after 20K iterations of L-BFGS training. The temperature is configured as (5) where $T_0(t) = [350 + 30 \times \sin(8 \times 10^{-8} \pi t)] K$. The relative errors shown in Table VI imply that the proposed STPINN method can achieve higher accuracy than PINN for solving the space–time-related diffusivity problem when the same number of neurons are employed. Fig. 17 shows the change process of objective function based on PINN, 1-SPINN, and 2-SPINN during the training phase. It can be observed that the loss of 2-SPINN is decreased to a lower value by employing the same number of neurons compared with 1-SPINN and PINN. Furthermore, 1-SPINN shows better generalization ability than PINN with a similar loss. It can be observed that, compared with the FEM, the mean relative error of the proposed STPINN decreases with an increased number of channels, whereas the time consumption increases with the broadening of the channel. For the proposed STPINN method, the number of channels should be determined by considering the accuracy and the time consumption.

In the traditional numerical methods, we need to adjust the spatial grid density according to the interconnect scale. Specifically, higher grid density means less relative error and more computational resources. Fig. 18 shows the improvement of computation time for the proposed STPINN method compared with the FEM under case III with 1K–15K meshing points along the interconnect wire. It can be seen from Fig. 18 that the speedup keeps rising when the number of meshing points is fewer than 9K. The STPINN method can achieve a speedup of around 40× when the number of meshing points is larger than 9K. As a result, the proposed mesh-free solver STPINN, not constrained by the number of meshing points for achieving higher accuracy, shows better computing performance on a more sophisticated stress analysis of VLSI interconnect wires.

We also compare the computation time between the proposed STPINN and FEM with almost the same accuracy. The smaller time step size is required for obtaining a more accurate numerical solution by the FEM, especially under complex thermal conditions. The stress values in the time range 0–10^8 s at the junction in case II obtained by
Multisegment Analysis

For multisegment analysis, the original self-diffusion coefficient \(D\). Multisegment Analysis calculating the whole stress values, respectively. It should be mentioned that the results obtained by the proposed STPINN method can be compared with those obtained by the FEM. It can be demonstrated that the proposed STPINN method shows the speedup of 9 \(\times\) with 10K iteration steps. Moreover, the proposed STPINN method shows the speedup of 9.30 \(\times\), 18.59 \(\times\), and 52.68 \(\times\) over the FEM with 100, 1K, and 10K iteration steps for calculating the whole stress values, respectively. It should be noted that the proposed STPINN method can achieve a 52.69 \(\times\) speedup while keeping a similarly high accuracy over the FEM.

D. Multisegment Analysis

We now extend the proposed STPINN method to the stress evolution calculation of interconnect wires with multisegments. For multisegment analysis, the original self-diffusion coefficient \(D_0\) is kept the same in each segment, which is used to calculate the diffusion coefficient \(D_s\) combining with space–time-related temperature. For the comparisons of FEM and STPINN, we choose a straight 7-segment interconnect wire and calculate the stress evolution under the dynamic temperature condition (case II) by the proposed 1-STPINN. We employ a 1-hidden layer network with 100 neurons per layer for \(F(x; \alpha)\) and a 7-hidden layer network with 50 neurons per layer in \(F(x; \beta)\) for calculating the stress values. Fig. 20(a) shows the current density distribution of this 7-segment interconnect tree. Fig. 20(b) shows the comparison results of the stress evolution in the time range from \(5 \times 10^7\) to \(1 \times 10^9\) s between the proposed method and the FEM. It can be demonstrated that the proposed STPINN method can accurately capture the stress evolution and the discontinuity through vias on the multisegment tree at different time instances.

Experimental results show that deepening and widening of the network architecture in \(F(x; \alpha)\) have little impact on the accuracy of stress evolution, which means that excessive calculation of time transformation is not necessary for solving EM-based stress evolution equation under the complex thermal condition. It can be also observed that for more complex topological structures governed by coupled stress evolution equations, a larger network structure for \(F(x; \beta)\) is required to approximate the stress evolution process.

E. Additional Input to the Proposed STPINN

We employ the EM driving force at the sample point as an additional input to STPINN. The solution of STPINN with an additional input can be expressed as \(F_s(x; G, F_t(x; \alpha); \beta; \theta)\). In the simulation of 1-STPINN, we choose a straight 19-segment interconnect tree and calculate the stress evolution under case II. We set \(F_t(x; \alpha)\) as the 1-hidden layer network with 100 neurons per layer and \(F(x; \beta)\) as the 8-hidden layer network with 40 neurons per layer. Fig. 21(a) shows the current density distribution of the interconnect tree and Fig. 21(b) shows the comparison results of the stress evolution in the time range from \(5 \times 10^7\) to \(5 \times 10^9\) s between 1-STPINN and FEM.

For obtaining the stress evolution of a cross-shaped 5-terminal interconnect, we employ a 2-hidden layer network with 50 neurons per layer in \(F_t(x; \alpha)\) and a 5-hidden layer network with 40 neurons per layer in \(F(x; \beta)\) for calculating the stress evolution under case III. The location input of the proposed 2-STPINN is a 2-D coordinate for the cross-shaped interconnect tree. The length and the current density of each segment are configured as \(L_1 = 20 \mu m, L_2 = 30 \mu m, L_3 = 10 \mu m, L_4 = 20 \mu m, j_1 = 1 \times 10^{10} \ A/m^2, j_2 = 2 \times 10^{10} \ A/m^2, j_3 = -3 \times 10^{10} \ A/m^2,\) and \(j_4 = 4 \times 10^{10} \ A/m^2\). The experimental results are shown in Fig. 22, which demonstrates good agreement between 2-STPINN and FEM. Compared with the number of neurons used in the 7-segment wire of Section IV-D, the additional input of STPINN decreases the demand for neurons in more complex interconnect structures. It demonstrates the importance of segment features in STPINN-based interconnect stress evolution analysis.

V. CONCLUSION

In this article, we propose a novel composite neural network to compute stress evolution along multisegment interconnect
trees during the void nucleation phase considering complex thermal conditions. The proposed STPINN method aims at obtaining mesh-free solutions of PDEs with space–time-related diffusivity. We first construct the interconnect thermal model for Joule heat spreading incorporating via effect and dynamic temperature. We then solve stress evolution equations under different temperature conditions by the STPINN method. To enhance the learning ability of neural network-based solver, space–time conversion and multichannels are used in the proposed STPINN method for obtaining a more accurate solution. The interior junction node constraints and the large gradients in the EM model are settled by the virtual distance and preprocessing techniques. Finally, we compare the stress evolution in configured interconnects under three different temperature configurations, which achieves mean relative errors < 1.22% versus FEM and < 0.44% versus the analytical model while delivering a $2 \times 52 \times$ speedup over the competing schemes. Of particular interest, the neural network-based large-scale EM stress approximation and the EM stochasticity assessment need further study. To obtain the stress evolution on large-scale interconnect trees through the neural network-based method, the key problem is how to efficiently reduce the total number of collocation points used for the physics-informed constraints. For EM stochasticity assessment which supposes that the self-diffusion coefficient in each segment is different, a more generalized learning-based model should be developed.

REFERENCES

[1] J. Warnock, “Circuit design challenges at the 14nm technology node,” in Proc. ACM/IEEE Des. Autom. Conf. (DAC), Jun. 2011, pp. 464–467.

[2] J. R. Black, “Electromigration—A brief survey and some recent results,” IEEE Trans. Electron Devices, vol. 16, no. 4, pp. 338–347, Apr. 1969.

[3] J. von Hagen, R. Bauer, S. Penka, A. Pietsch, W. Walter, and A. Zitzelsberger, “Extrapolation of highly accelerated electromigration tests on copper to operation conditions,” in Proc. IEEE Int. Integ. Rel. Workshop Final Rep., 2002, pp. 41–44.

[4] R. L. de Orio, H. Ceric, and S. Selberherr, “Physically based models of electromigration: From black’s equation to modern TCAD models,” Microelectron. Rel., vol. 50, no. 6, pp. 775–789, 2010.

[5] S. X.-D. Tan, M. Tahoori, T. Kim, S. Wang, Z. Sun, and S. Kiamher, Long-Term Reliability of Nanometer VLSI Systems: Modeling Analysis and Optimization, Cham, Switzerland: Springer, 2019.

[6] H. Zheng, B. Yin, K. Zhou, L. Chen, and C. Kuo, “Temperature-dependent activation energy of electromigration in copper low-k interconnects,” J. Appl. Phys., vol. 122, no. 7, 2017, Art. no. 74501.

[7] V. Mishra and S. S. Sapatnekar, “The impact of electromigration in copper interconnects on power grid integrity,” in Proc. ACM/IEEE Des. Autom. Conf. (DAC), May 2013, pp. 1–6.

[8] X. Huang, A. Kteyan, S. X.-D. Tan, and V. Sukharev, “Physics-based electromigration models and full-chip assessment for power grid networks,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 35, no. 11, pp. 1848–1861, Nov. 2016.

[9] L. Chen, S. X.-D. Tan, Z. Sun, S. Peng, M. Tang, and J. Mao, “Fast analytic electromigration analysis for general multisegment interconnect wires,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 28, no. 2, pp. 421–432, Feb. 2020.

[10] V. Sukharev, A. Kteyan, and X. Huang, “Postvoiding stress evolution in confined metal lines,” IEEE Trans. Device Mater. Rel., vol. 16, no. 1, pp. 50–60, Mar. 2016.

[11] S. Chatterjee, V. Sukharev, and F. N. Najm, “Fast physics-based electromigration assessment by efficient solution of linear time-invariant (LTI) systems,” in Proc. IEEE/ACM Int. Conf. Comput.-Aided Des. (ICCAD), 2017, pp. 659–666.

[12] S. Chatterjee, V. Sukharev, and F. N. Najm, “Power grid electromigration checking using physics-based models,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 37, no. 7, pp. 1317–1330, Jul. 2018.

[13] X. Wang et al., “Fast physics-based electromigration analysis for full-chip networks by efficient eigenfunction-based solution,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 40, no. 3, pp. 507–520, Mar. 2021.

[14] Z. Sun, E. Demircan, M. D. Shroff, T. Kim, X. Huang, and S. X.-D. Tan, “Voltage-based electromigration immortality check for general multi-branch interconnects,” in Proc. IEEE/ACM Int. Conf. Comput.-Aided Des., 2016, pp. 1–7.

[15] X. Huang, T. Yu, V. Sukharev, and S. X.-D. Tan, “Physics-based electromigration assessment for power grid networks,” in Proc. 51st Annul. Des. Autom. Conf., 2014, pp. 1–6.

[16] M. A. Korhonen, P. Borgesen, K. N. Tu, and C.-Y. Li, “Stress evolution due to electromigration in confined metal lines,” J. Appl. Phys., vol. 73, no. 8, pp. 3790–3799, 1993.

[17] H.-B. Chen, S. X.-D. Tan, X. Huang, T. Kim, and V. Sukharev, “Analytical modeling and characterization of electromigration effects for multi-branch interconnected trees,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 35, no. 11, pp. 1811–1824, Nov. 2016.

[18] C. Michoski, M. Milosavljević, T. Oliver, and D. R. Hatch, “Solving differential equations using deep neural networks,” Neurocomputing, vol. 399, pp. 193–212, Jul. 2020.

[19] J. Sirignano and K. Spiliopoulos, “DGM: A deep learning algorithm for solving partial differential equations,” J. Comput. Phys., vol. 375, pp. 1339–1364, Dec. 2018.

[20] M. W. M. G. Dissanayake and N. Phan-Thien, “Neural-network-based approximations for solving partial differential equations,” Commun. Numer. Methods Eng., vol. 10, no. 5, pp. 195–201, 1994.

[21] I. E. Lagaris, A. C. Likas, and D. G. Papageorgiou, “Neural-netw‐rk methods for boundary value problems with irregular boundaries,” IEEE Trans. Neural Netw., vol. 11, no. 5, pp. 1041–1049, Sep. 2000.

[22] I. E. Lagaris, A. C. Likas, and D. I. Fotiadis, “Artificial neural networks for solving ordinary and partial differential equations,” IEEE Trans. Neural Netw., vol. 9, no. 5, pp. 987–1000, Sep. 1998.

[23] L. Ruthotto and E. Haber, “Deep neural networks motivated by partial differential equations,” J. Math. Imag. Vis., vol. 62, pp. 352–364, Apr. 2020.

[24] M. Raissi and G. E. Karniadakis, “Hidden physics models: Machine learning of nonlinear partial differential equations,” J. Comput. Phys., vol. 357, pp. 125–141, Mar. 2018.

[25] Y. Zang, G. Bao, X. Ye, and H. Zhou, “Weak adversarial networks for high-dimensional partial differential equations,” J. Comput. Phys., vol. 411, Jun. 2020, Art. no. 109409.

[26] M. W. M. G. Dissanayake and N. Phan-Thien, “Neural-network-based approximations for solving partial differential equations,” Commun. Numer. Methods Eng., vol. 10, no. 5, pp. 195–201, 1994.

[27] W. Jin, S. Peng, and S. X.-D. Tan, “Data-driven electrostatics analysis based on physics-constrained deep learning,” in Proc. Des. Autom. Test Eur. Conf. Exhibit. (DATE), 2021, pp. 1382–1387.

[28] M. Raissi, P. Perdikaris, and G. E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” J. Comput. Phys., vol. 378, pp. 686–707, Feb. 2019.

[29] A. G. Baydin, B. A. Pearlmutter, A. A. Radul, and J. M. Siskind, “Automatic differentiation in machine learning: A survey,” J. Mach. Learn. Res., vol. 18, no. 1, pp. 5595–5637, 2018.
[30] L. Lu, X. Meng, Z. Mao, and G. E. Karniadakis, “DeepXDE: A deep learning library for solving differential equations,” *SIAM Rev.*, vol. 63, no. 1, pp. 208–228, 2019.

[31] S. P. Hau-Riege and C. V. Thompson, “The effects of the mechanical properties of the confinement material on electromigration in metallic interconnects,” *J. Mater. Res.*, vol. 15, no. 8, pp. 1797–1802, 2000.

[32] V. Sukharev and F. N. Najm, “Electromigration check: Where the design and reliability methodologies meet,” *IEEE Trans. Device Mater. Rel.*, vol. 18, no. 4, pp. 498–507, Dec. 2018.

[33] X. Huang, V. Sukharev, J.-H. Choy, T. Kim, and S. X.-D. Tan, “Electromigration assessment for power grid networks considering temperature and thermal stress effects,” *Integration*, vol. 55, pp. 307–315, Sep. 2016.

[34] M. Chew, A. Aslyan, J.-H. Choy, and X. Huang, “Accurate full-chip estimation of power map, current densities and temperature for em assessment,” in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Des. (ICCAD)*, 2014, pp. 440–445.

[35] T.-Y. Chiang, K. Banerjee, and K. C. Saraswat, “A new analytical thermal model for multilevel ULSI interconnects incorporating via effect,” in *Proc. IEEE Int. Interconnect Technol. Conf.*, 2001, pp. 92–94.

[36] T.-Y. Chiang, K. Banerjee, and K. C. Saraswat, “Effect of via separation and low-k dielectric materials on the thermal characteristics of cu interconnects,” in *Int. Electron Devices Meeting Tech. Dig. (IEDM)*, 2000, pp. 261–264.

[37] K. Banerjee, A. Amerasekera, G. Dixit, and C. Hu, “The effect of interconnect scaling and low-k dielectric on the thermal characteristics of the IC metal,” in *Int. Electron Devices Meeting Tech. Dig.*, 1996, pp. 65–68.

[38] M. Raissi, A. Yazdani, and G. E. Karniadakis, “Hidden fluid mechanics: A Navier-Stokes informed deep learning framework for assimilating flow visualization data,” *Science*, vol. 367, no. 6481, pp. 1026–1030, 2020.

[39] X. Meng, Z. Li, D. Zhang, and G. E. Karniadakis, “PFNN: Parareal physics-informed neural network for time-dependent PDEs,” *Comput. Methods Appl. Mech. Eng.*, vol. 370, Oct. 2020, Art. no. 113250.

[40] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural Netw.*, vol. 2, no. 5, pp. 359–366, 1989.

[41] M. Stein, “Large sample properties of simulations using Latin hypercube sampling,” *Technometrics*, vol. 29, no. 2, pp. 143–151, 1987.

[42] A. N. F. Verspyt and R. D. Braatz, “Analysis of finite difference discretization schemes for diffusion in spheres with variable diffusivity,” *Comput. Chem. Eng.*, vol. 71, pp. 241–252, Dec. 2014.

[43] “COMSOL Multiphysics.” [Online]. Available: https://www.comsol.com (Accessed: Feb. 12, 2021).

[44] H.-B. Chen, S. X.-D. Tan, J. Peng, T. Kim, and J. Chen, “Analytical modeling of electromigration failure for VLSI interconnect tree considering temperature and segment length effects,” *IEEE Trans. Device Mater. Rel.*, vol. 17, no. 4, pp. 653–666, Dec. 2017.

[45] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.

[46] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, “A limited memory algorithm for bound constrained optimization,” *SIAM J. Sci. Comput.*, vol. 16, no. 5, pp. 1190–1208, 1995.

**Tianshu Hou** received the B.S. degree in electronic information science and technology from Sichuan University, Chengdu, Sichuan, China, in 2019. She is currently pursuing the Ph.D. degree with the Department of Micro/Nano-Electronics, Shanghai Jiao Tong University, Shanghai, China.

Her current research interests include electromigration reliability modeling, assessment, and optimization.

**Tianshu Hou**

**Ngai Wong** (Senior Member, IEEE) received the B.Eng. and Ph.D. degrees in electrical and electronic engineering from The University of Hong Kong (HKU), Hong Kong, in 1999 and 2003, respectively.

He was a Visiting Scholar with Purdue University, West Lafayette, IN, USA, in 2003. He is currently an Associate Professor with the Department of Electrical and Electronic Engineering, HKU. His research interests include electronic design automation, model order reduction, tensor algebra, linear and nonlinear modeling and simulation, and compact neural network design.

**Ngai Wong**

**Quan Chen** (Member, IEEE) received the B.S. degree in electrical engineering from Sun Yat-sen University, Guangzhou, China, in 2005, and the M.Phil. and Ph.D. degrees in electronic engineering from The University of Hong Kong (HKU), Hong Kong, in 2007 and 2010, respectively.

From 2010 to 2011, he was a Postdoctoral Fellow with the Department of Computer Science and Engineering, University of California at San Diego, San Diego, CA, USA. From 2012 to 2018, he was a Research Assistant Professor with the Department of Electrical and Electronic Engineering, HKU. He joined the Southern University of Science and Technology, Shenzhen, China, in 2019, where he is currently an Assistant Professor. His research interests include ultralarge-scale circuit simulation and multiphysics analysis in the field of electronic design automation (EDA), as well as EDA techniques for emerging technologies, such as sub-10-nm devices, memristors, and quantum computing. He has also years of experience in technical transformation and commercialization.

**Quan Chen**

**Zhigang Ji** received the Ph.D. degree from Liverpool John Moores University, Liverpool, U.K., in 2010.

He is currently a Professor of Nanoelectronics with Shanghai Jiao Tong University, Shanghai, China. His current research interests focus on characterizing, modeling and design of reliable, low-power, and high-performance computation systems, motivated by both evolutionary and revolutionary advances in nanoelectronics.

**Zhigang Ji**

**Hai-Bao Chen** (Member, IEEE) received the B.S. degree in information and computing sciences and the M.S. and Ph.D. degrees in applied mathematics from Xi’an Jiaotong University, Xi’an, China, in 2006, 2008, and 2012, respectively.

He then joined Huawei Technologies, Shenzhen, China, where he focused on cloud computing and big data. He was a Postdoctoral Research Fellow with the Electrical Engineering Department, University of California at Riverside, Riverside, CA, USA, from 2013 to 2014. He is currently an Associate Professor with the Department of Micro/Nano-Electronics, Shanghai Jiao Tong University, Shanghai, China. He has authored or coauthored about 70 papers in scientific journals and conference proceedings. His current research interests include very large-scale integration (VLSI) reliability, machine learning and neuromorphic computing, numerical analysis and modeling for VLSIs, and integrated circuit for signal and control systems.

**Hai-Bao Chen**

**Dr. Chen** received one Best Paper Award nomination from Asia and South Pacific Design Automation Conference in 2015. Since 2016, he has been serving as an Associate Editor for *Integration, the VLSI Journal*.