Machine Learning Approach for Fast Electromigration Aware Aging Prediction in Incremental Design of Large Scale On-Chip Power Grid Network

SUKANTA DEY, Indian Institute of Technology Guwahati, INDIA
SUKUMAR NANDI, Indian Institute of Technology Guwahati, INDIA
GAURAV TRIVEDI, Indian Institute of Technology Guwahati, INDIA

With the advancement of technology nodes, Electromigration (EM) signoff has become increasingly difficult, which requires a considerable amount of time for an incremental change in the power grid (PG) network design in a chip. The traditional Black's empirical equation and Blech's criterion are still used for EM assessment which is a time-consuming process. In this paper, for the first time, we propose a machine learning approach to obtain the EM-aware aging prediction of the PG network. We use neural network-based regression as our core machine learning technique to instantly predict the lifetime of a perturbed PG network. The performance and accuracy of the proposed model using Neural Network are compared with the well-known standard regression models. We also propose a new failure criterion based on which the EM-aging prediction is done. Potential EM-affected metal segments of the PG network is detected by using a logistic-regression based classification machine learning technique. Experiments on different standard power grid benchmarks show a significant speedup for our machine learning model compared to the state-of-the-art models. The predicted value of MTTF for different power grid benchmarks using our approach is also better than some of the state-of-the-art MTTF prediction models and comparable to the other accurate models.

CCS Concepts: • **Hardware** → *Package-level interconnect*; *Power grid design*; *Metallic interconnect*; *Aging of circuits and systems*; *Failure prediction*; *Economics of chip design and manufacturing*; • **Computing methodologies** → *Neural networks*; *Classification and regression trees*;

Additional Key Words and Phrases: Electromigration, Machine Learning, MTTF, Neural Network, Power Grid Network, Regression, Reliability

ACM Reference Format:
Sukanta Dey, Sukumar Nandi, and Gaurav Trivedi. 2020. Machine Learning Approach for Fast Electromigration Aware Aging Prediction in Incremental Design of Large Scale On-Chip Power Grid Network. *ACM Trans. Des. Autom. Electron. Syst.* 25, 5, Article 42 (May 2020), 29 pages. https://doi.org/https://doi.org/10.1145/3399677

1 INTRODUCTION

Electromigration (EM) has become one of the major reliability issues for the interconnects of the newer technology nodes [29, 37, 40]. As metal interconnects suffer most due to the EM-based reliability issues, the power grid (PG) network of a VLSI Chip is highly susceptible to EM, activated by the momentum transfer of free electrons to the metal atoms. While the signal and clock line also

Authors' addresses: Sukanta Dey, Dept. of Computer Science and Engineering, Indian Institute of Technology Guwahati, Guwahati, Assam, 781039, INDIA; Sukumar Nandi, Dept. of Computer Science and Engineering, Indian Institute of Technology Guwahati, Guwahati, Assam, 781039, INDIA; Gaurav Trivedi, Dept. of Electronics and Electrical Engineering, Indian Institute of Technology Guwahati, Guwahati, Assam, 781039, INDIA.

1084-4309/2020/5-ART42 $15.00
https://doi.org/https://doi.org/10.1145/3399677

suffer from EM degradation, these lines carry bidirectional current and which results in a longer lifetime due to the so-called *healing* effects. However, power grid interconnects mostly carry a unidirectional current which has no privilege of healing effect and as a result more susceptible to EM degradation.

The standard practice in the industry is to use traditional Black's model [4] for all the interconnects of the PG network and then considering the earliest branch failure time as the lifetime of the entire grid which takes hours of time. For successful EM sign off, EM violations of the interconnects need to be checked iteratively by changing the design incrementally in every step. Therefore, to speedup the EM sign off process, incremental analysis of the EM is necessary. Recently many works on physics-based EM model is proposed for optimistic aging prediction of the PG network. Huang et al. [19] have proposed such a physics-based model for the first time. Mishra et al. [26] proposed a better approach for predicting the lifetime of the PG network considering transient stress modeling. Chatterjee et al. [9] proposed a fast physics-based electromigration assessment using an efficient solution of linear time-invariant systems. Wang et al. [38] proposed a physics-based model using integral transformation technique. Chatterjee et al. [10] extended their work on LTI system-based EM assessment approach by incorporating macromodeling-based filtering and predictor approach . There are several other works on physics-based EM assessment model [11, 36, 39]. Dey et al. [14] proposed that minimization of IR drop which results in an optimistic EM-aging prediction of VLSI PG network. However, most of these methods still take hours of time to obtain the lifetime of the chip as it involves solving partial differential equations (PDE). Practical adaptation of these physics-based methods [9–11, 19, 28, 36, 38, 39] for a full scale EM prediction for a chip is not possible as these methods are time-consuming. Although, recent work of Najm and Sukharev [30] shows a significant speedup for full chip simulation in EM-aging prediction by employing Monte-Carlo Simulation. However, the work of [30] still requires a fresh EM-simulation for incremental changes in the design. Therefore, in order to speed up the EM-sign off phase of the incremental PG network design and to facilitate a practical method which can be adapted for full scale EM prediction, it is better to reuse the historical data (generated by the standard aging models) and use these data to create a machine learning model which can instantly predict the EM-aging of the PG network.

Motivation of Machine Learning Approach for EM-Aging Prediction: Recently, EM-aging has emerged as the design problem [3]. The traditional practice of determining EM-based aging is time-consuming. If there is an incremental change or any small perturbations in the power grid design during its design phase, then to check for the acceptable EM margin once more takes hours of time using conventional EM-aging computation models. As machine learning (ML) has been proved to be an effective approach for predicting tasks over the last few decades, this approach can be used to predict the EM-aging of the PG network, which would take very little time to obtain EM-aging of the PG network compare to the traditional approaches. In literature, there are few developments of ML-based approach. Dey et al. [15] proposed a learning-based model for on-chip power grid design. Huang et al. [20] used a machine learning approach in detection and classification of defects in TSV-based 3D IC. Elfadel et al. [16] have reviewed many applications of VLSI CAD using Machine Learning. In our machine learning approach, the model has to be trained just once for a certain power grid topology and subsequently it can predict the EM-aging for small perturbations in the PG network with very little testing time (CPU runtime). Therefore, in this paper, using a machine learning approach we predict the EM-aging of the PG network of a chip by constructing a regression-based model. Our method uses the historical data generated from the traditional time-consuming EM-aging computation model to learn the EM-aging of the PG Network and eventually able to predict the EM aging time. Our main motivation of this paper is to demonstrate that the EM-aging prediction of the PG network can be done using Machine learning approach. Experiments on different PG benchmarks showed that using the historical data of EM-aging model for our Machine Learning approach reduces the overall EM sign-off time significantly in comparison to all the state-of-the-art models [9, 10, 19, 30] and the predicted MTTF is also found to be better than that of [9, 10, 30] and comparable to [19].

Novel contribution: To the best of our knowledge, this paper is the first to use the Machine Learning approach to predict the EM-aging in incremental design of the VLSI PG network. The major contributions of our proposed work are:

- Fast KLU solver is used for power grid analysis, from which currents and voltages of the power grid networks are obtained. These values are utilized in feature extraction and training data generation phase.
- Feature selection is made evaluating the r^2 score of different features. Accordingly, the training dataset is generated using the power grid benchmarks, and the Black's model is used for labeling the training data. Test data set (which is different from the training data set) is generated by perturbing the same benchmarks data.
- EM-aging prediction problem of the VLSI PG network is formulated as a regression-based supervised machine learning model by selecting different features (related to the properties of EM).
- Different regression-based Machine Learning models have been used to obtain the best model for EM-aging prediction by evaluating the model accuracy metrics and other performance metrics.
- A new failure criterion has been proposed analytically based on the worst-case IR drop of the PG network, which is utilized with the machine learning model to predict the EM-aging of the PG network.
- The proposed approach for EM-aging prediction using the machine learning model can predict the EM-aging for the test data set. The accuracy of the EM-aging model is demonstrated by changing different test data sets with a variation of perturbation size.
- Further, a logistic-regression based classification model is used to obtain the potentially weak EM affected metal segments of the PG network.
- Experimental results on different power grid benchmarks show that the proposed EM-aging prediction approach using ML is faster than all the state-of-the-art models [9, 10, 19, 30]. The predicted value of MTTF is also found to be better than that of [9, 10, 30] and comparable to [19]. This proves the efficiency of the proposed ML-based in the EM-aging prediction model in the incremental design of the VLSI power grid network.

The rest of the paper is organized as follows. In Section 2, EM fundamentals and PG network model used in this paper have been discussed. Section 3 contains the problem formulation for the aging prediction, training data generation, and the proposed machine learning model is described. Section 4 describes the utilization of the ML approach for EM-aging prediction. This section also describes a new failure criterion of the PG network. Identification of the potentially EM-affected degraded metal segment using a logistic regression-based classification technique is explained in this section. Different experiments on the standard power grid benchmarks are carried out to validate our proposed method using machine learning in Section 5. The paper is concluded in Section 6.

2 BACKGROUNDS

2.1 Electromigration fundamentals

Electromigration is the process of movement of metal atoms due to the exchange of momentum from the electrons to the metal atoms. The EM degradation can happen in two phases: *void nucleation* and *void growth*. Under high current in the metal lines, metal atoms are subjected to stress for a prolonged period of time, which causes the void to occur. This EM degradation phase is termed as *void nucleation* phase. Once the void nucleates, it started to grow which is termed as *void growth* phase. The aging of the metal lines due to EM degradation is measured as *mean-time-to-failure* (MTTF). Black [4] has proposed an empirical equation to evaluate the MTTF of the metal interconnects due EM, which is given as follows,

$$MTTF = \frac{A}{J^n} e^{\frac{E_a}{kT}}, \tag{1}$$

which evaluates the interconnect MTTF based on known current density (J) and temperature (T). A is a constant depends on the metal geometry, grain size, and current density. The value of n is found to be 2 which, E_a is the EM activation energy, and k is the Boltzman's constant. Blech [5] observed that the mortality of the metal interconnects vary with the length. He proposed a criterion for the filtration of immortal interconnects, which follows,

$$(JL) \leq (JL)_c = \frac{\Omega \sigma_c}{eZ\rho} \tag{2}$$

L is the length of the metal interconnect, Ω is the atomic volume, e is the electron charge, eZ is the effective charge of the migrating atoms, ρ is the resistivity of the metal interconnect, σ_c is the critical stress needed for the failure of the metal interconnect. Korhonen et al. [23] proposed a mathematical formulation to represent the *hydrostatic stress* σ which originates from the influence of EM.

$$\frac{\partial \sigma}{\partial t} = \frac{\partial}{\partial x}\left[c\left(\frac{\partial \sigma}{\partial x} + \frac{eZ\rho J}{\Omega}\right)\right], \tag{3}$$

where $c = \frac{D_a B \Omega}{kT}$, where D_a is the atomic diffusivity, B is the bulk modulus. Approximate value of the nucleation time can be obtained from (3). A void nucleates once the stress exceeds the critical value.

Physics-based Models: The basic idea of the physics-based models is solving the interconnect trees[1], which are generated from the power grid netlist. Initially, the current densities of all the interconnect trees are calculated. Subsequently, PDEs associated with the interconnects (refer (3)) are solved in order to find the stress level. The solution of the PDE gives us the nucleation time (the time required for nucleation[2]) for critical values of stress. This nucleation time dominates the MTTF value. In [19], the authors use an iterative method that looks after the changes in stress and the power grid resistance as a function of time. With time the stress level, as well as the resistance of the power grid network, exceeds a critical value, and the simulation stops. The cumulative nucleation time is considered as MTTF of the power grid network. Power grid analysis is done in every iteration to observe the voltage drop level (V_{ir}) of the interconnects, which acts as the stopping criteria of the simulation. If the voltage drop level reaches above a threshold level (V_{th}), the power grid is considered dysfunctional, and the lifetime is calculated. This process is described pictorially in Fig. 1. As it is an iterative process, it takes a large amount of computational time to converge. Therefore, there is a need to have new methods for EM aging evaluation, which can converge very fast.

2.2 Power Grid Model

EM is a long-term phenomenon which considers the average effects of the currents, therefore, a DC load model of the PG network (Fig. 2) with only resistive elements of the metal line is considered [8]. From the steady-state model of the PG network, the system of linear equations is obtained

[1] Interconnect trees: connected graph of the interconnects.

[2] Nucleation: first stage of formation of a void.

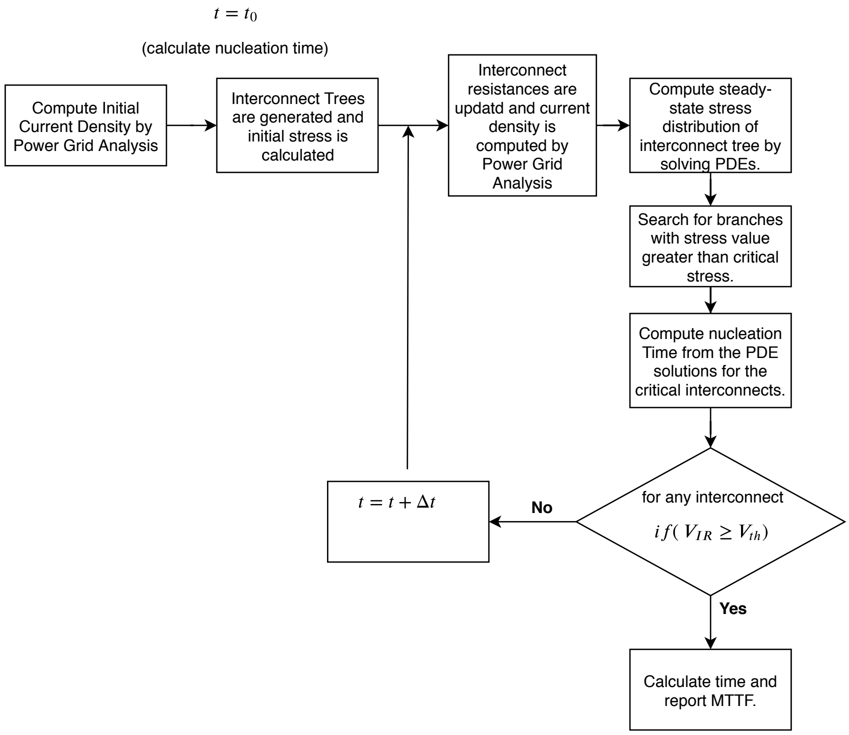

$t = t_0$
(calculate nucleation time)
Compute Initial Current Density by Power Grid Analysis
Interconnect Trees are generated and initial stress is calculated
Interconnect resistances are updatd and current density is computed by Power Grid Analysis
Compute steady-state stress distribution of interconnect tree by solving PDEs.
Search for branches with stress value greater than critical stress.
Compute nucleation Time from the PDE solutions for the critical interconnects.
for any interconnect
$if(V_{IR} \geq V_{th})$
No
$t = t + \Delta t$
Yes
Calculate time and report MTTF.

Fig. 1. Basic Idea of Physics-based EM model

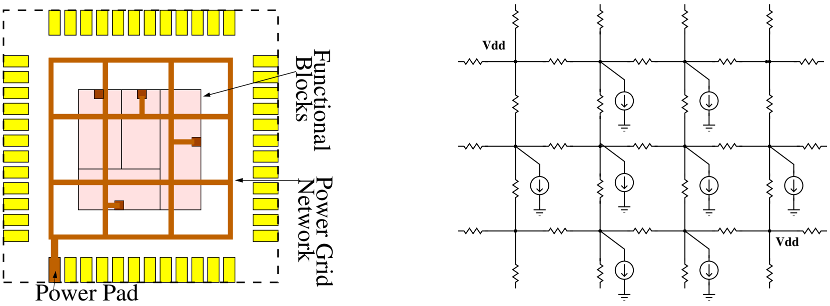


Fig. 2. (a) Floorplan of a VLSI SoC with its PG network connecting the functional blocks. (b) The resistive DC load model of the PG network.

which can be written as **GV** = **I**, where **G** is the conductance matrix, **V** is the node voltages vector, and **I** is the current sources vector connected to grounds. Even though the probabilistic solvers [13, 33] and locality-based solvers [21, 24] have gained a lot of attention in the recent past. Direct solvers have always been the first choice for solving the system of linear equations. In this work, we use KLU-based direct solver [12] to obtain the node voltages vector **V**. Furthermore, using all the node voltages of the PG network, branch currents can be obtained. The hydrostatic stress of all the interconnects can be obtained by using (3) under a given applied current condition. Now if we perform integration of (3), we obtain the expression of σ:

$$\sigma(t,x) = \sigma_0 + \frac{\partial}{\partial x}\left[\int_0^t c\frac{\partial\sigma}{\partial x}dt + c\frac{eZ\rho}{\Omega}\int_0^t Jdt\right] \tag{4}$$

which describes that the hydrostatic stress distribution of an interconnect depends on the time integral of the applied current density. This can be considered as a justification of substituting the current waveforms with the time averaged DC current. Therefore, effective current density of the interconnect metal lines can be represented by [25],

$$J = \frac{1}{T}\left(\int_0^T J^+(t)dt - \psi\int_0^T |J^-(t)|dt\right), \tag{5}$$

where ψ is the EM recovery factor and found experimentally, $J^+(t)$ is the current density from one side of the wave. For the unidirectional current of the PG network, the effective-current density is the time-averaged current density.

3 PROPOSED MACHINE LEARNING MODEL

3.1 Problem Formulation

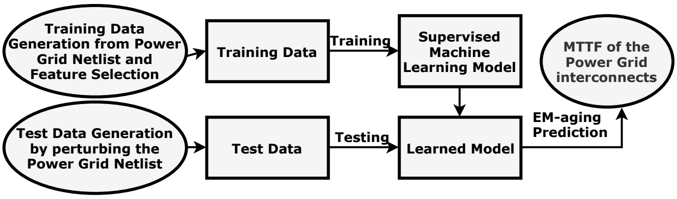

Training Data Generation from Power Grid Netlist and Feature Selection
Training Data
Training
Supervised Machine Learning Model
MTTF of the Power Grid interconnects
Test Data Generation by perturbing the Power Grid Netlist
Test Data
Testing
Learned Model
EM-aging Prediction

Fig. 3. Flow of Machine Learning Model for EM-aging Prediction

The flow of our supervised machine learning model for EM-aging prediction is shown in Fig. 3. The input to our supervised machine learning model for EM-aging prediction is a set of n training samples of interconnects of a PG network, where each training sample i has m *input features* (denoted as X_i) and one *output feature* (denoted as y_i). The objective is to train our model with the input features and then to predict the output of a new test sample τ (denoted as y'_τ) by a function of only the input features. For our aging prediction model, the output feature is the MTTF of the interconnects of the PG network. The input features are related to the properties of EM for different interconnects of the PG network. Regression is the process of constructing the relationship between independent variables (input features) and dependent variables (output features) in order to estimate the output of the dependent variables. In the simple regression model, we only consider a single input feature for prediction. To make the prediction correct, we can consider different input features which is considered as multiple regression. In our proposed machine learning model, we have considered a few input features of the interconnect of the PG network to demonstrate the effectiveness of the EM-aging prediction model. We have used the neural network as our core machine learning technique since this technique is proved to be the most effective for the supervised learning problem.

3.2 Feature Selection and Training Data Generation

As shown in Fig. 4, we know that the MTTF depends on many parameters. However, it has been observed that MTTF changes significantly with the variation of current density (J), temperature (T), and the length of the interconnect (L). Hence current density, temperature, and interconnect

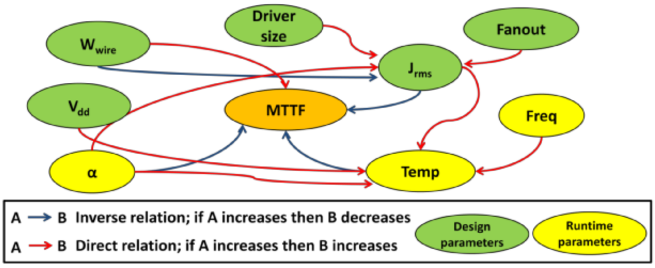

Driver size
W$_{wire}$
Fanout
J$_{rms}$
V$_{dd}$
MTTF
Freq
α
Temp
A → B Inverse relation; if A increases then B decreases
A → B Direct relation; if A increases then B increases
Design parameters
Runtime parameters

Fig. 4. Design and Runtime parameters affecting EM MTTF [22].

length can be considered as the input features of our aging prediction machine learning model. Dey et al. [14], showed that variation of IR drop of the metal lines result in a variation of MTTF of the PG network, henceforth, IR drop of all the interconnects can also be considered as an input feature. Therefore, current density, length of the interconnect and IR drop of the interconnects considered as input features for our machine learning model for aging prediction. The coefficient of determination (r^2 score) [27] between the MTTF and different input features is listed in Table 1. r^2 score denotes the proportion of variance of the MTTF (output feature) that is predictable from the input features. A higher value of $r^2 (\leq 1)$ is desired for selecting the best combination of input features for the model. This is the main intention of using r^2 score for feature selection so that we know which model fits properly for our EM-aging dataset. The r^2 score of Table 1 demonstrates that the combination of J, L, IR drop, and T as the input features (as multiple regression) result in a more accurate regression model which predicts MTTF value nearest to the actual value. For the purpose of completeness, we are defining the r^2 score definition for our EM-aging prediction model as given in Definition 1.

Definition 1. ***(r^2 score)*** *The coefficient of determination (or r^2 score) between the MTTF (output feature) and different input features denote the proportion of variance of the MTTF (output feature) that is predictable from the input features is defined as follows,*

$$r^2 = 1 - \frac{SS_{residual}}{SS_{total}}, \tag{6}$$

where $SS_{residual} = \sum_i \left(y_{\tau_i} - y'_{\tau_i}\right)^2$ is residual sum of squares and $SS_{total} = \sum_i \left(y_{\tau_i} - \bar{y}_\tau\right)^2$is total sum of squares. Here, y_{τ_i} denotes actual value of the i^{th} test sample, y'_{τ_i} denotes predicted value of i^{th} test sample, and $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_{\tau_i}$. A larger value of $r^2 (\leq 1)$ is desirable for the data of the input features and output feature to perfectly fit in the regression model.

Proposition 1. *The supervised machine learning model for EM-aging prediction fits best as multiple regression model with the combination of input features J, L, IR drop and T.*

Proof. From the Definition 1, we know that a higher value of r^2 score is desirable for the best fit of the data for the EM-aging model. A simple expreiment to find r^2 score with different combination of input features show that for the combination of J, L, IR drop and T, the r^2 score is highest as listed in the Table 1. Therefore, we can say that our EM-aging model fits best for the combination of input features J, L, IR drop and T. □

Proposition 1 describes about the feature selection for our EM-aging model. More about the EM-aging model property with different input features can be found in the ablation study of the model (Please refer to Appendix). The training data containing the input features and output

Table 1. r^2 score between MTTF vs different input features

Input features	J	L	IR drop	T	J, L, T, and IR drop combined
r^2 score	0.638	0.753	0.859	0.781	0.978

feature is generated using the Algorithm 1 for different power grid benchmarks. Initially, power grid analysis is done using KLU solver [12] to obtain all the branch currents of the PG network. Subsequently, IR drop of all the interconnect is obtained and finally using the Black's [4] series method MTTF for all the interconnects is obtained. For finding the temperature (T) of the PG interconnects, we have used the thermal model as proposed in [41]. In this way, for a PG network, we have generated the training set which is used to train the regression model. The flow of the machine learning model for EM-aging prediction is shown in Fig. 3. We briefly discussed IR drop analysis using KLU Solver, as it is an important part of the training data generation phase.

IR drop analysis using KLU Solver: At first, the power grid circuit in the form of the SPICE netlist is feed as input to the EM aging prediction framework. IR drop analysis is done in order to obtain the currents and voltages of the power grid circuit. Subsequently, using the modified nodal analysis(MNA), the system of linear equations is solved, which gives us the currents and the voltages of the power grid circuit. KLU solver [12] is employed for solving the system of linear equations, as it has a more considerable speedup than the HSPICE circuit simulator [1], which is demonstrated in Table 5. The better speedup of KLU is obtained, as it efficiently determines the solution of the linear system of equations resulting from the modified network analysis. The matrix is permuted in block triangular form in KLU solver, and each block is ordered to reduce the fill. Gilbert-Peierls algorithm is employed for performing the LU factorization, and then the system is subsequently solved using block back substitution [12]. Once the power gird circuit is solved using the KLU Solver, and all circuit parameters such as current and voltages are determined. These parameters are used for obtaining the training dataset of the power grid circuit.

ALGORITHM 1: Training Data Generation

```
Input: Power grid netlist.
Output: Training dataset.
Interconnect length ($L$) is extracted from netlist;
Power grid analysis is done using KLU solver to find the currents of all the interconnects.;
J is calculated for the interconnects;
IR drop of each of the interconnect is obtained;
Temperature ($T$) of each of the interconnect is calculated using the thermal model [41];
Using Black's model, MTTF of each of the interconnects of the PG network is obtained;
```

3.3 Proposed ML Model using Neural Network Regression

Neural network is a non-linear function which takes some input features and gives some output feature as shown in Fig.5. These input and output layers of a neural network form an arrangement of connected layers, where each layer contains a few neurons represented by nodes. There are three types of layers in a neural network i.e., one input layer, one output layer, and a number of hidden layers. The number of neurons in the input layer depends on the number of input features of the training dataset. For solving the regression problem, the number of neuron in the output layer is only one. There can be various numbers of the hidden layers and the number of neurons on each of the hidden layers can vary. The main aim of the neural network is to optimize the weights of each of the connections of the neurons in order to reduce the error cost function J_e.

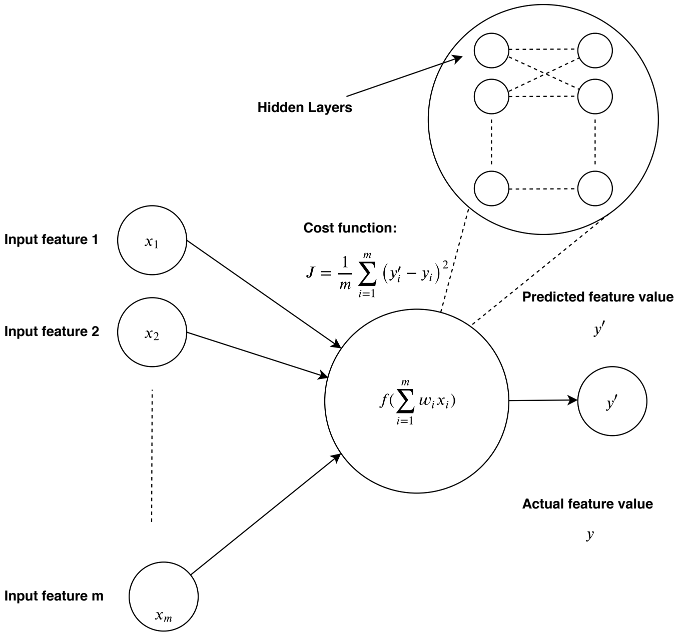


Fig. 5. An illustration of neural network with its input features and output feature with magnified view of the hidden layers.

For our EM-aging model, we considered four neurons in the input layer as we have considered four input features. Each of these four features have n training samples. We apply bias in each of the layers in order to generalize the model. We apply activation function in all the nodes which is represented as $a_u^{[l]}$ refers to the activation function of the u^{th} neuron unit in the layer l, where for the input layer $l = 0$. Activation functions are important, as the characteristics possed by it is the assumed to be the characteristic of the neuron. For each neuron node, one activation function is applied. The four input features of the EM aging prediction model can be represented as follows:

$$x_1 = a_1^{[0]} \tag{7}$$

$$x_2 = a_2^{[0]} \tag{8}$$

$$x_3 = a_3^{[0]} \tag{9}$$

$$x_4 = a_4^{[0]}, \tag{10}$$

The output of the hidden layers are given as follows as mentioned in [32],

$$z_u^{[l]} = W_u^{[l]^T} x + b_u^{[l]} \tag{11}$$

$$a_u^{[l]} = g(z_u^{[l]}) \tag{12}$$

where, l is the l^{th} hidden unit number, and u is the u^{th} neuron in the l^{th} hidden unit. Here b represents the bias applied in each layer, W represents the weight matrix, and z represents the combination of b and W, which is further given as input to the activation function. The output of the final output layer is given as follows:

$$z_u^{[l]} = W_u^{[l]^T} a^{[l-1]} + b_u^{[l]} \tag{13}$$

$$a_u^{[l]} = g(z_u^{[l]}) \tag{14}$$

We have used rectilinear unit (RelU) activation function, whose response is as the following:

$$g(z) = \max(z, 0), \tag{15}$$

where $g(z)$ is a non-linear function. Using the activation of the output layer i.e., (14) we can get the value of the MTTF predicted by Neural Network. Once the predicted value is acheived, cost function is formulated in order to increase prediction accuracy. We have used the mean squared cost function which is given below,

$$\text{Minimize} \;\; J_e = \frac{1}{m} \sum_{i=1}^{m} \left(y_i' - y_i\right)^2 \tag{16}$$

Using any optimizer such as Adam optimizer or gradient descent approach, the cost function is optimized and the weights of the neural network are updated. In this way, accurate values of the output feature is obtained.

3.4 Test Data Generation and Incremental Analysis

For the purpose of test data generation we have perturbed a region of the power grid network. Perturbation is done using three ways:

1. By varying the node voltages in a region: A region with $\gamma_1\%$ nodes of the PG network are considered for perturbation and the voltages of each of the $\gamma_1\%$ nodes are changed by an amount of 1% of V_{dd}.

2. By varying the current workload in a region: Similarly, a region with $\gamma_2\%$ current sources of the PG network are considered for perturbation and the current values of each of the $\gamma_2\%$ current sources are changed by an amount of 1% of the maximum current of the PG network.

3. By varying both the node voltages and current workload in a region: In this case, a region with $\gamma_1\%$ of nodes of the PG network and $\gamma_2\%$ current sources of the PG network are considered for perturbation. Voltages of each of the $\gamma_1\%$ nodes are changed by an amount of 1% of V_{dd} and the current values of each of the $\gamma_2\%$ current sources are changed by an amount of 1% of the maximum current of the PG network.

Perturbed power grid network is solved in a faster way using the incremental power grid analysis method as done by Boghrati et al. [6]. We have adapted our KLU solver for the incremental power grid analysis to speed up the test data generation process. The adapatation comes as follows,

$$\begin{aligned} (G + \Delta G)(V + \Delta V) &= (I + \Delta I) \\ \Rightarrow (G + \Delta G)\Delta V &= \Delta I - \Delta G V \\ \Rightarrow G_{eff}\Delta V &= \Delta I_{eff} \end{aligned} \tag{17}$$

where $G_{eff} = (G + \Delta G)$ and $I_{eff} = \Delta I - \Delta G V$. As dimensions of ΔV and ΔI_{eff} are very small. Hence, by solving (17), we get the perturbed values of the power grid network instantly. We don't require to perform a fresh power grid analysis again which is time consuming. In this way the test dataset is generated for validating our EM-aging ML model.

ALGORITHM 2: MTTF Prediction using Proposed ML Model

```
Input: Training dataset.
Output: MTTF of the entire PG network.
Train the model for Aging prediction;
if $JL \geq (JL)_c$ then
    Predict MTTF using EM-aware aging prediction model for the interconnects of the test data set;
end
MTTF of the first $\eta$% mortal interconnect is considered as MTTF of the PG network as proposed in
Theorem 1, 2.
```

4 EM ASSESSMENT USING ML MODEL

4.1 EM-Aware Aging Prediction using Proposed ML Model

Machine Learning based aging prediction model has been illustrated in Algorithm 2. Initially, the training data set is generated for different power grid benchmarks using the Algorithm 1. As the numerical value of different features varies significantly in magnitude, therefore, normalization of all the features is done in order to fit the training data properly into the regression model. Subsequently, the aging prediction model is constructed using Neural Network-based regression technique. Activation is applied in all the nodes of the Neural Network and the cost function using mean-squared-error is constructed. Adam optimizer is used to minimize the cost function. Accurately predicted values are obtained for the output feature MTTF. Therefore, for any given training features (X_i, y_i) and also the test features (X_τ, y'_τ) a non-linear function $f(\sum_{i=1}^{m} w_i x_i)$ can be formed in terms of the weights of the connections of the different neurons (w_i) and the input features(x_i) with the help of the activation function. In this way, once the regression model is trained using a large number of samples of training data (X_i, y_i), then the model is able to predict y'_τ for any new unseen sample X_τ. The prediction accuracy of the EM-model is evaluated using the metric mean-square-error (MSE) as described next.

Mean Square Error (MSE) is defined as the average squared difference between the estimated values using EM-aging ML model and the true values of the MTTF (output features) of all the interconnect samples of the PG network. A value of MSE closer to and greater than zero is desirable for the model to have high accuracy.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_{\tau_i} - y'_{\tau_i})^2 \tag{18}$$

y_{τ_i} is the actual value of the output feature of i^{th} sample of the test dataset and y'_{τ_i} is the predicted value of the output feature of i^{th} sample of the test dataset.

We obtain the predicted MTTF value from the EM-aging ML model. However, in order to achieve an MTTF value comparable to the state-of-the-art results, we propose a new failure criterion which is described next.

4.2 Proposed Failure Criterion For The PG Network

Chatterjee et al. [8] showed that mortality of first interconnect does not ensure the aging of the PG network as there must be some alternative current carrying path to the mortal affected line. Huang et al. [19] in their work proposed that the failure criterion for the PG Network depends on the worst case IR drop noise of the PG Network. We have analytically obtained a direct relation between the worst-case IR drop noise, and expected number of mortal interconnects as established in Theorem 1. From which it can be deduced that the PG Network becomes dysfunctional while the expected number of mortal interconnects is η% of the total interconnects which is more clearly elaborated in the Theorem 2. Our aim is also to obtain an optimistic prediction of MTTF of the PG Network. Therefore, in order to obtain an optimistic MTTF prediction of the PG network, we have used this new failure criterion of the PG Network. With the help of EM-aging prediction model, MTTF of the first $\eta\%$ interconnects which become mortal is calculated and is considered as the MTTF of the PG network as given in the Algorithm 2.

Theorem 1. *The worst case IR drop noise of the PG Network (caused by change in resistance due to EM) goes above V_{th}, if and only if the expected number of mortal interconnects due to EM is approximately $\eta\%$ of the total interconnects.*

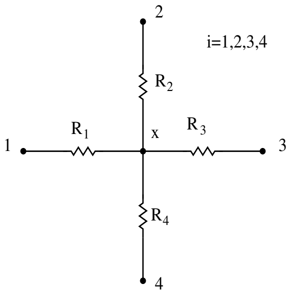


Fig. 6. Nodes directly connected to the worst case IR drop noise node x

Proof. Let the total number of interconnects be M. Let the total number of nodes be N. Let we have $q(\leq N)$ instances of the worst-case IR drop node. Let $K_x(\leq M)$ be the total number of interconnect directly connected to the node having the worst-case IR drop noise x. If the IR drop noise across any interconnect (V_{IR}) connected to node x goes above a certain threshold voltage V_{th} (i.e., $V_{IR} \geq V_{th}$) then we assume that the interconnect will fail or become mortal. The probability of the i^{th} interconnect (connected to node x) to fail, $P_i(V_{IR} \geq V_{th}) = \frac{v_x - v_i}{\sum_{i=1}^{K}(v_x - v_i)}$, where v_x and v_i are node voltages of the node x and i respectively (refer Fig. 6). The probability of any interconnect to be connected with a worst-case IR drop noise = $q.\frac{\binom{N}{1}}{\binom{N}{2}}$. The probability of the i^{th} interconnect to be connected with a worst-case IR drop noise and also to fail= $q.\frac{\binom{N}{1}}{\binom{N}{2}} \sum_{x=1}^{q} \frac{v_x - v_i}{\sum_{i=1}^{K_x}(v_x - v_i)}$. The estimated total number of interconnects (β) which are connected with a worst-case IR drop noise and also to fail is,

$$\beta = M.q.\frac{\binom{N}{1}}{\binom{N}{2}} \sum_{x=1}^{q} \frac{v_x - v_i}{\sum_{i=1}^{K_x}(v_x - v_i)}. \tag{19}$$

$$\Rightarrow \beta = \eta\% \text{ of } M \tag{20}$$

where

$$\eta = 100.q.\frac{\binom{N}{1}}{\binom{N}{2}} \sum_{x=1}^{q} \frac{v_x - v_i}{\sum_{i=1}^{K_x}(v_x - v_i)} \tag{21}$$

Therefore, the expected number of mortal interconnects due to EM is $\eta\%$ of total interconnects. Similarly, if we assume that total mortal interconnects is $\eta\%$ of M, then it can be obtained that the worst case IR drop noise of the PG network goes above V_{th}. □

Theorem 2. *The PG Network is considered to be dysfunctional if the worst-case IR drop noise goes above V_{th} [35], which is equivalent to the fact that PG Network can also be considered to be dysfunctional if the expected number of mortal interconnects is approximately $\eta\%$ of the total interconnects.*

Proof. From the Theorem 1, it can be concluded that the worst case IR drop noise of the PG Network directly depends on the expected number of mortal interconnects of the PG Network. The failure criterion as proposed in [35] states that the PG Network becomes dysfunctional once the worst case IR drop noise goes above V_{th}. Hence, from the Theorem 1, it can be deduced that the PG Network becomes dysfunctional while the expected number of mortal interconnects is approximately $\eta\%$ of the total interconnects. □

We have shown an example of a sample PG network in Example 1, in order to see a practical values of expected number of mortal interconnects as depicted in the Theorem 1.

Example 1. *For a PG network with 9 nodes and 12 edges as shown in Fig. 7, the expected number of mortal interconnect is $\eta \approx 14$ as found by using the Theorem 1. $V_{dd} = 1.8V$, All resistors have 0.5Ω and all current source values are $0.05A$.*

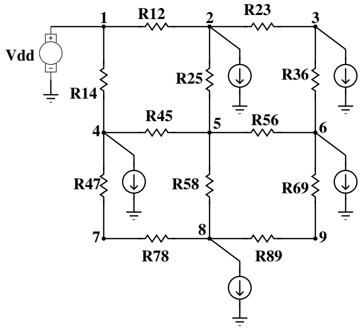


Fig. 7. A PG network with 9 nodes and 12 edges

Proof. Here, N= 9, M=12, and let us consider $q = 1$ since the worst case IR drop noise occurs across $R14$ resistor in the Fig. 7 which is obtained by PG analysis. By doing PG analysis, we have got voltages of all the nodes as follows, $V_1 = 1.8V$, $V_2 = 1.5750V$, $V_3 = 1.5656V$, $V_4 = 1.6750V$, $V_5 = 1.6094V$, $V_6 = 1.5813V$, $V_7 = 1.6406V$, $V_8 = 1.6063V$, $V_9 = 1.5983V$.

Now, using (21), we get the value of η as follows,

$$\eta = 100.q.\frac{\binom{9}{1}}{\binom{9}{2}} \frac{V_{R14}}{V_{R14} + V_{R45} + V_{R47}} \tag{22}$$

$$= 100.1.\frac{\binom{9}{1}}{\binom{9}{2}} \frac{125mV}{125mV + 65mV + 34mV} \tag{23}$$

$$= 13.95 \tag{24}$$

Therefore, the number of mortal interconnects are,

$$\beta = \eta\% of M \tag{25}$$

$$= 13.95\% of 12 \tag{26}$$

$$= 1.674 \approx 2 \tag{27}$$

From this example we come to know that the PG network of Fig. 7 becomes dysfunctional when the two interconnects of the network becomes mortal. This analytical expression of η is used to find the MTTF of the PG network.

□

4.3 MTTF Prediction

The test set is generated by perturbing the same power grid benchmarks which are used for training data generation for validating the EM-aging ML model. Perturbation of the PG network is done for $\gamma_1\%$ = $\gamma_2\%$= 10% as mentioned in the previous section for all the three cases. For the perturbed PG network, a similar procedure as mentioned in the Algorithm 1 (except for the output feature MTTF) is used to generate the test data set. Our trained model based on the Neural Network is tested using these test sets to predict MTTF of the PG network as mentioned in Algorithm 2. For the failure criterion of the PG network using our proposed model, we have selected MTTF of the first $\eta\%$ mortal interconnect as the MTTF of the PG network as mentioned in Theorem 1 and Theorem 2. To show the accuracy of the proposed model with the variation in the test set size, we compare the prediction accuracy with different test sets by varying the perturbation size ($\gamma_1\%$ = $\gamma_2\%$) by 10%, 25%, 40%, 55%, 70%, 85% and 100% respectively for all the three cases (refer Fig. 10).

4.4 Identification of EM-affected Metal Segment of PG Network

Once the EM-aging prediction is completed, then it is necessary to detect the potential EM-affected metal segments of the PG network in order to obtain the reliable design of the PG network. For the identification of the potentially degraded metal segment, all the metal segments are numbered chronologically and labeled as mortal or non-mortal in the dataset based on the MTTF of the PG Network as determined by our proposed method mentioned in the previous section. We can also obtain the mortal interconnects in this stage of the simulation flow. However, if in the incremental design phase of the PG network, we want to create a model for identification of the mortal interconnects then we need to label the dataset and further create another machine learning-based classification model.Therefore, in view of this, a machine learning-based classification problem is formulated for classifying the degraded metal segments of PG Network. To accomplish this, we have used logistic regression which is a binary classification technique. We have used theses labeled dataset in order to train the classification model. The potential EM-affected metal segments can be identified using this binary classification technique from the trained model. The efficiency of the classification can be obtained using metrics such as precision, recall, and F1-score. These classification metrics are defined in the Table 2. For detecting potential EM-affected mortal PG interconnects, true positive means mortal interconnect is predicted as mortal, and true negative means mortal non-mortal interconnect is predicted as non-mortal. Similarly, if mortal interconnect is predicted as non-mortal then it is called false positive, and if non-mortal interconnect is predicted as mortal then it is called as false negative. After the prediction using logistic regression-based classification, from the chronological numbers of the interconnects we obtain the predicted mortal interconnects. Subsequently, these interconnect designs are changed in order to make the power grid design EM resilient.

5 EXPERIMENTAL RESULTS

5.1 Simulation Setup

All the experiments are done using C++, and Python programming languages on a 3.2 GHz Linux based machine with 32 GB memory. For different machine learning models we have used scikit-learn [2] machine learning library. To test our proposed approach IBM power grid benchmark [31]

Table 2. Accuracy Metrics and its definition for logistic regression-based classification for detecting potentially EM-affected mortal PG interconnects.

Metrics	**Expression**
Accuracy	$\frac{\text{True Positive+True Negative}}{\text{True Positive+True Negative+Flase Positive+Flase Negative}}$
Precision	$\frac{\text{True Positive}}{\text{True Positive+Flase Positive}}$
Recall	$\frac{\text{True Positive}}{\text{True Positive+Flase Negative}}$
F1-Score	$2\times\frac{\text{Precision}*\text{Recall}}{\text{Precision+Recall}}$

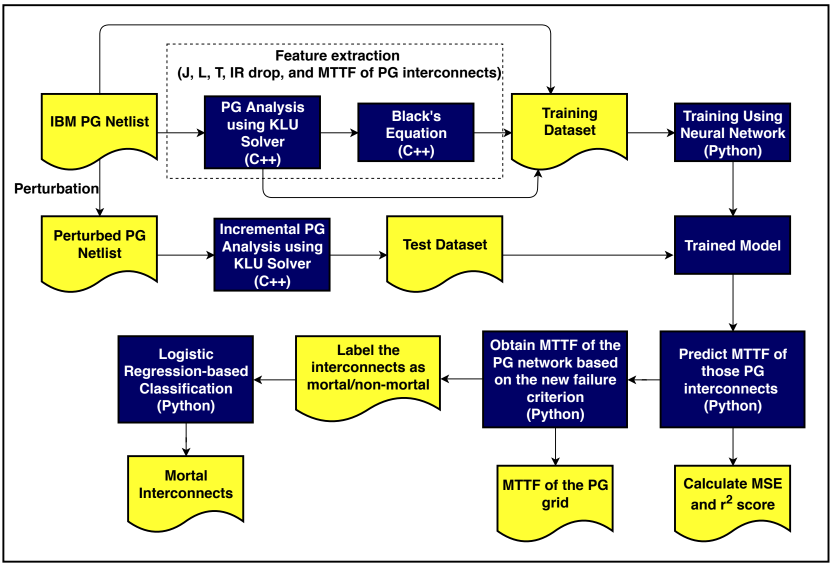

Feature extraction
(J, L, T, IR drop, and MTTF of PG interconnects)
IBM PG Netlist
PG Analysis using KLU Solver (C++)
Black's Equation (C++)
Training Dataset
Training Using Neural Network (Python)
Perturbation
Perturbed PG Netlist
Incremental PG Analysis using KLU Solver (C++)
Test Dataset
Trained Model
Logistic Regression-based Classification (Python)
Label the interconnects as mortal/non-mortal
Obtain MTTF of the PG network based on the new failure criterion (Python)
Predict MTTF of those PG interconnects (Python)
Mortal Interconnects
MTTF of the PG grid
Calculate MSE and r^2 score

Fig. 8. Flow of the simulation setup of the Machine Learning Flow

and industry-based in-house power grid benchmarks are used. These PG benchmark statistics are mentioned in Table 4. The total simulation flow is shown in Fig. 8. Initially, from the power grid netlist feature extractions are done to extract J, L, T, IR drop, and MTTF of the PG interconnects. Power grid analysis using KLU Solver [12] is performed in order to obtain the currents and voltages of the power grid circuit and subsequently Black's equation is used. From which all the features are obtained and the dataset is prepared. Eventually, we train the neural network-based regression model for the model to learn the behavior of the dataset. For testing purpose, we have done the incremental power grid analysis which is a faster method. It facilitates the change in the current and voltages of the perturbed netlist in a faster way. Finally, the test dataset is tested on the neural network-based learned model and we get the predicted MTTFs of the PG interconnects. For each of the PG circuit, we have obtained r^2 score and MSE in order to verify the testing accuracy of the models. Once we get the predicted MTTFs of PG interconnects, we apply the new failure criterion of the PG network, to find the MTTF of the PG network. Simultaneously, we have labeled the interconnects as mortal or non-mortal depending on its MTTFs and then applied a logistic regression-based supervised learning technique to detect the potential mortal interconnects. All the results recorded in this section are averaged over 10 trails of each of the experiments.

Neural Network Architecture: The neural network architecture and the hyperparameters used in the experiments are described here. We have used ten hidden layers and Rectified Linear Unit (ReLU) activation function in our neural network architecture. Other hyperparameters of the neural network are listed in Table 3. The hyperparameters and the number of hidden layers are obtained after performing an exhaustive grid search using the `GridSearchCV` tool of the scikit-learn library. The notations used in Tabel 3 are standard notations used for regression models in scikit-learn library [2].

Table 3. Hyperparameters used in the Neural Network Architecture

Items	**Value**
#hidden layers	10
Activation	ReLU
Solver	Adam
α	0.001
batch size	Auto
learning rate	Constant
initial learning rate	0.01
power_t	0.5
max_iter	1000
momentum	0.9
validation fraction	0.1
β_1	0.9
β_2	0.999
ϵ	1e-08

Table 4. IBM PG benchmark [31] and industry-based PG benchmarks statistics

PG Circuits	**#n**	**#r**	**#v**	**#i**
*PG*1	10001	19800	200	9801
*ibmpg*1	30638	30027	14308	10774
*ibmpg*2	127238	208325	330	37926
*ibmpg*3	851584	1401572	955	201054
*ibmpg*4	953583	1560645	962	276976
*ibmpg*5	1079310	1076848	539087	540800
*ibmpg*6	1670494	1649002	836239	761484
*ibmpgnew*1	1461036	2352355	955	357930
*ibmpgnew*2	1461039	1422830	930216	357930
*PG*2	1000001	1998000	2000	998001
*PG*3	4000001	7996000	4000	3996001
*PG*4	9000001	17994000	6000	8994001

#n : total number of vertices (nodes) of PG network, **#r** : total number of resistors (edges) of PG network, **#v** : total number of supply voltage (V_{dd} and *GND* supply source) of PG network, **#i** : total number of current workloads connected to PG network.

Our power grid analysis method using KLU Solver is faster than the industry standard *Synopsys HSPICE circuit simulator*[1]. The speedup of KLU solver over HSPICE is listed in Table 5. The reason behind the better speedup is KLU efficiently solves the linear system of equations resulting from the modified network analysis. KLU solver is used for circuit simulation in the feature selection phase. We have also used this KLU solver for incremental PG analysis in the test data generation phase.

Table 5. Comparison of circuit analysis time for HSpice and KLU Solver

	Time (sec)		**Speedup**
PG circuits	**HSPICE**	**KLU Solver**	$\frac{\textbf{Time}_{HSPICE}}{\textbf{Time}_{KLU}}$
$PG1$	0.90	0.15	6.00×
$ibmpg1$	2.85	0.56	5.08×
$ibmpg2$	19.46	2.61	7.45×
$ibmpg3$	29.60	5.07	5.73×
$ibmpg4$	54.4	5.83	9.33×
$ibmpg5$	73.80	7.74	9.53×
$ibmpg6$	96.5	10.10	9.55×
$ibmpg\_nw1$	103.58	14.50	7.14×
$ibmpg\_nw2$	47.60	8.86	5.37×
$PG2$	67.14	12.85	5.22×
$PG3$	247.36	49.80	4.96×
$PG4$	587.60	120.37	4.88×

5.2 Total Time Required for the Machine Learning Model

Table 6. Training and Testing time for power grid benchmarks using Neural Network

PG circuits	t_1 **(s)**	t_2 **(s)**	t_3 **(s)**	t_4 **(s)**	t_{ML} **(s)**	t_{ML} **(min)**	t_{ML} **(hr)**
$PG1$	0.30	0.001	0.069	0.001	0.371	0.006	0.0001
$ibmpg1$	1.10	0.001	0.697	0.002	1.800	0.021	0.0003
$ibmpg2$	5.20	0.01	1.088	0.006	6.304	0.105	0.002
$ibmpg3$	10.25	0.02	21.378	0.179	31.827	0.530	0.009
$ibmpg4$	11.33	0.02	12.476	0.165	23.991	0.390	0.007
$ibmpg5$	15.62	0.03	8.810	0.096	24.196	0.400	0.006
$ibmpg6$	21.52	0.03	13.174	0.147	34.871	0.580	0.009
$ibmpgnew1$	28.46	0.04	18.920	0.220	47.640	0.790	0.013
$ibmpgnew2$	16.78	0.03	13.079	0.129	30.018	0.500	0.008
$PG2$	25.47	0.03	10.611	0.132	36.243	0.600	0.010
$PG3$	100.45	0.05	50.663	0.676	151.839	2.530	0.042
$PG4$	245.89	0.19	117.216	1.557	364.853	6.080	0.101

[*] t_1 = training data generation time including power grid analysis time using KLU solver and Black's model implementation time in seconds, t_2 = test data generation time including incremental power grid analysis time in seconds, t_3 = training time in seconds, t_4 = testing time including prediction time in seconds, $t_{ML} = t_1 + t_2 + t_3 + t_4$, t_{ML} (s) = total time required summing all the times in secs, t_{ML} (min) = t_{ML} converted in minutes, t_{ML} (hr) = t_{ML} converted in hour.

The training data generation time including the circuit simulation time for the different IBM power grid benchmarks is given in Table 6. Our main aim is to train the data of a certain PG network and then to predict the MTTF of the PG network for any incremental changes in the PG design. For the training data generation, we used the KLU solver to solve the $GV = I$ matrix. However, for the test data generating, we have used the incremental analysis using KLU solver for the incremental PG design, to speed up the process as mentioned in Section 3.4. To show that we have done the perturbations in the PG network design and then predict the MTTF by our trained ML-model.

5.3 Results of Expected number of Mortal interconnects of PG Network

The expected number of mortal interconnects (η of (21)) of the PG network until the PG network is considered dysfunctional is calculated using for all the PG benchmarks using the Theorem 1. We consider three instances of worst-case IR drop while evaluating the value of η by varying the value

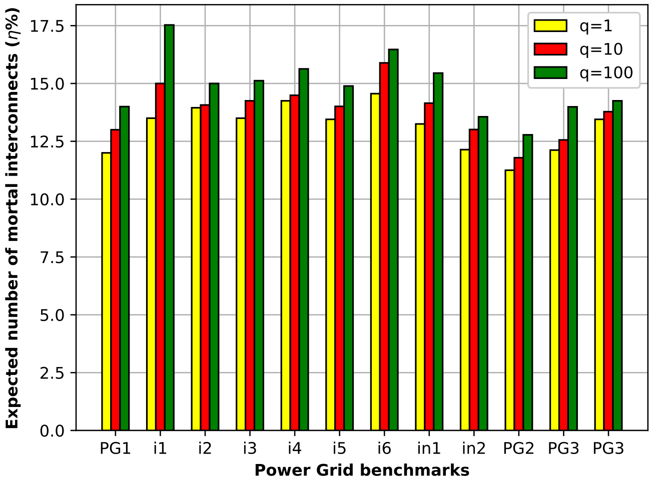


Fig. 9. Expected number of mortal interconnects (η% of total PG interconnects) for different PG benchmarks until the PG network is considered dysfunctional. Here $i1$, $i2$, $i3$, $i4$, $i5$, $i6$ $in1$, and $in2$ denote $ibmpg1$, $ibmpg2$, $ibmpg3$, $ibmpg4$, $ibmpg5$, $ibmpg6$, $ibmpgnew1$, and $ibmpgnew2$ benchmark respectively.

of q. It is observed that as the value of q increases from 1 to 100, the number of mortal interconnects until the failure of the PG network increases. The results of η for different PG benchmarks are shown in Fig. 9. This result is shown in order to observe the practical values of η for the PG benchmarks.

5.4 Results of MTTF Prediction and CPU Runtime

Table 7. Comparison of MTTF for our proposed ML-based approach with works of [9, 10, 19, 30] for IBM power grid benchmarks.

	MTTF (μ) (years)				
Methods	**TCAD2016** [19] (μ_H)	**ICCAD2017** [9] (μ_{Ch})	**TCAD2018** [10] (μ_C)	**IRPS2019** [30] (μ_N)	**Proposed** (μ_{ML})
PG Circuits					
*PG*1	14.01	6.10	8.51	6.5	13.25
*ibmpg*1	12.55	6.50	10.91	7.0	12.10
*ibmpg*2	18.75	6.78	10.11	12.1	12.55
*ibmpg*3	31.96	6.66	9.95	6.7	12.25
*ibmpg*4	33.39	9.83	11.95	16.7	17.48
*ibmpg*5	25.16	6.54	6.63	6.3	10.33
*ibmpg*6	19.87	9.53	11.96	11.2	12.41
*ibmpgnew*1	25.96	13.24	11.64	13.2	14.56
*ibmpgnew*2	21.80	5.72	6.72	7.3	13.24
*PG*2	17.85	8.32	9.32	10.3	11.21
*PG*3	-	-	-	7.2	10.51
*PG*4	-	-	-	6.8	8.47

This section demonstrates the improvement in the predicted value of MTTF and CPU runtime of the EM aging prediction model. The results of MTTF and CPU runtime for all the power grid benchmarks are listed in Table 7 and Table 8 respectively. We have compared our results with the

Table 8. Comparison of CPU Runtime for our proposed ML-based approach with works of [9, 10, 19, 30] for IBM power grid benchmarks.

	CPU Runtime (t) (Hours)					**Speedup**			
Methods	**TCAD2016** [19] (t_H)	**ICCAD2017** [9] (t_{Ch})	**TCAD2018** [10] (t_C)	**IRPS2019** [30] (t_N)	**Proposed** (t_{ML})	$\frac{t_H}{t_{ML}}$	$\frac{t_{Ch}}{t_{ML}}$	$\frac{t_C}{t_{ML}}$	$\frac{t_N}{t_{ML}}$
PG Circuits									
*PG*1	0.02	0.02	0.001	0.000166	0.0001	200×	200×	10×	1.66×
*ibmpg*1	0.05	0.03	0.003	0.01000	0.0003	166.66×	100×	10×	33.33×
*ibmpg*2	0.11	0.31	0.04	0.02000	0.002	55×	155×	20×	10×
*ibmpg*3	5.83	4.27	0.41	0.07000	0.009	647.77×	610×	45.55×	7.77×
*ibmpg*4	14.71	6.81	2.31	0.11000	0.007	2101.42×	972.85×	330×	15.71×
*ibmpg*5	0.69	0.25	0.06	0.03000	0.006	115×	41.66×	10×	5×
*ibmpg*6	1.75	2.07	0.79	0.23330	0.009	194.44×	230×	87.77×	25.92×
*ibmpgnew*1	16.78	0.42	1.24	0.08000	0.013	1290.76×	32.06×	95.38×	6.15×
*ibmpgnew*2	15.32	2.60	0.43	0.06000	0.008	1915×	325×	53.75×	7.50×
*PG*2	10.94	1.12	1.06	0.10166	0.010	1094×	112×	106×	10.06×
*PG*3	-	-	-	0.13666	0.04200	-	-	-	3.25×
*PG*4	-	-	-	0.25666	0.10100	-	-	-	2.54×
Avg. Speedup						778×	277.85×	76.84×	10.74×

works of Chatterjee et al. [9, 10], Huang et al. [19] and Najm et al. [30], which are also listed in Table 7 and Table 8. We have got a significant improvement in the speedup for our machine learning based EM-aging prediction approach over the work of [9, 10],[19], and [30] as listed in the Table 8. The reason for this significant speedup of our ML-based approach is that the time taken for training dataset generation, to train the model, and then predicting the MTTFs for the test dataset is much lesser than those physics-based state-of-the-art models. The physics-based models solve the PDE (corresponding to the interconnects) and solve the equations for the whole power grid network to obtain the MTTF of the power grid network. Generally, the solutions are obtained using iterative PDE solver or some approximation techniques. As a result, the execution time takes a considerable amount of time to converge. In [19], the authors adopted a similar methodology, as shown in Fig. 1. The model of [19] is one of the most accurate EM evaluation models, as the authors analytically solve the PDEs. However, this approach takes a huge time. For large PG circuits, this model doesn't converge due to limitations of the system memory. In [9], authors have extended the existing physics-based models for EM evaluation and represented as linear time-invariant (LTI) systems. This LTI system is solved iteratively to get the MTTF. This method also takes a considerable amount of time and system memory. In [10], authors further extended their LTI system-based EM assessment approach by incorporating macromodeling-based filtering and predictor approach. However, in our ML-based EM-aging model, we train the model with historical data, and then testing is done using a test dataset, which does not take much time. As a result, we get significant speedups compared to [9, 10, 19] (Please refer to Table 8). The results of [9, 10, 19] are reproduced using the similar EM model parameters as reported in [9, 10, 19]. The MTTF prediction results using our machine learning approach is also better than that of [9, 10] and comparable to the accurate physics-based model of [19]. Our predicted MTTF values are better than [9, 10] and comparable to [19], because of our new EM-failure criterion which we discussed in Section 4.

Due to the large size of the PG3 (~4M nodes and ~7.9M interconnects) and PG4 (~9M nodes and ~17.9M interconnects) circuits, the models of [9, 10, 19] need huge system memory and don't converge on our system with 32 GB memory and 3.2 GHz processor. All the three models [9, 10, 19] generate interconnect trees from the PG netlist. Subsequently, the solution is obtained by solving the PDEs corresponding to interconnect trees iteratively. Generating the solution of PDEs for large PG circuits makes the MTTF computation expensive in terms of execution time and system memory. As a result, our implementation of the physics-based models [9, 10, 19] does not converge on our system for the PG3 and PG4 circuits. This also shows the scalability of our proposed machine learning model for large scale PG circuits.

Recent work by Najm et al. [30] shows a significant speedup over the work of [9, 10, 19] for MTTF prediction of PG network. They have used Monte-Carlo random sampling for predicting the MTTF of a large scale PG network. We also obtained the results using EM models of [30] and listed it in Table 7 and Table 8. From the results, it can be observed that our proposed approach also outperforms the results of [30] in terms of both speedup and EM lifetime (our EM lifetime is much closer to the EM lifetime of [19]). Therefore, it can be proved that the CPU runtime of EM-aging prediction using our machine learning method is faster than the work of [30]. Even the value of MTTF predicted by our EM-aging model with the new failure criterion gives an improvement in MTTF than that of [30]. Therefore, deploying this machine learning approach decreases the EM sign off time significantly with an estimated error (MSE) of less than 3% for a perturbation size of 10% (see Figure 10). More about MSE and perturbation size are given in the following results.

It is worth mentioning that the work [19], which was published in 2016, presented an accurate method. However, the method of [19] is time and computing resource-consuming and not feasible to adapt in real-world full-chip EM lifetime prediction of large-scale power grid circuit. Please note that the work [9, 10, 30], which was subsequently published in 2017, 2018, 2019 were not accurate and have significant speedup compared to [19]. Our proposed ML-based approach is faster than [9, 10, 30], and EM lifetime results are much closer to [19] than that of [9, 10, 30]. With our proposed approach, it will be easier for the designers to get an approximate estimation of the EM lifetime in very less time, which makes the total design cycle faster.

5.5 Results of Regression Model Prediction Accuracy

This experiment is done to show the comparison of accuracy of the proposed EM-aging prediction model using Neural Network (NN) over the other well-known regression techniques which includes Bayesian linear regression [27], Random Forest regression [18], Ridge linear regression [17], SVM regression [7], Gaussian Process Regression (GPR) [34] are shown in this section. The metrics used to compare the accuracy are the r^2 score and the mean-square-error (MSE) for all the test sets generated from the IBM power grid benchmarks[31]. A higher value r^2 score (≤ 1) denotes better compactness of the data to the regression model. The r^2 score and MSE value of the test set for all the power grid benchmarks are listed in the Table 9, and Table 10 respectively. Simultaneously, the results are also compared with all other standard regression techniques with their best possible settings for prediction. The results show that for all the seven IBM power grid benchmarks, the Neural Network model outperformed other models (see Neural Network column in Table 9 and Table 10). Although, the GPR regression technique gives a better value of r^2 score and MSE for two benchmark circuits. However, GPR needs huge runtime memory requirement and hence it is not able to give any feasible result for the larger PG benchmarks (*ibmpg*2 onwards). From this experiment, we want to prove that our EM-aging model for PG network using the Neural Network works best compared to the other standard regression models in terms of r^2 score and MSE value.

5.6 Comparison of MSE with Variations in Perturbation Size

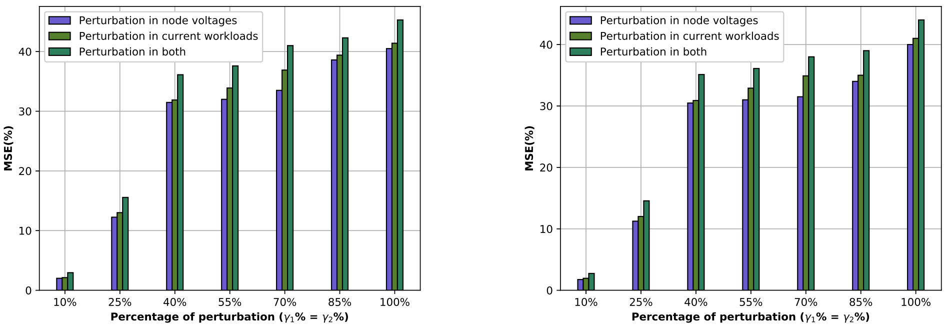
This section demonstrates the test accuracy of the EM-prediction model in the incremental design of the PG network by varying the perturbation size of the test set. For the incremental design of the power grid network, usually, the changes or the perturbations done in designs in each iterative step is much less than 10% [6]. Therefore, for all the experiments throughout the manuscript, we have kept the perturbation size constant at 10%. However, in order to see the test accuracy for larger perturbation, in this section, we have used different perturbation size from 10%, 25%, 40%, 55%, 70%, 85%, and 100% and verify the MSE for different PG benchmarks data. Results of the comparison of

Table 9. MSE of the EM aging models tested by different regression techniques For IBM power grid benchmarks.

PG Circuits	#interconnects	Mean Square Error					
		Bayesian	Random Forest	Ridge	SVM	Gaussian	Neural Network
*PG*1	10001	3.69%	5.97%	3.71%	8.35%	2.42%	2.99%
*ibmpg*1	30027	3.66%	5.93%	3.61%	8.23%	2.32%	2.95%
*ibmpg*2	208325	4.25%	6.24%	3.92%	10.45%	-	3.11%
*ibmpg*3	1401572	4.52%	7.56%	4.21%	12.12%	-	2.74%
*ibmpg*4	1560645	3.95%	6.67%	3.75%	9.25%	-	2.89%
*ibmpg*5	1076848	4.14%	5.92%	3.83%	9.72%	-	3.31%
*ibmpg*6	1649002	3.93%	6.54%	3.89%	10.37%	-	3.20%
*ibmpgnew*1	2352355	4.23%	6.89%	3.75%	8.52%	-	3.15%
*ibmpgnew*2	1422830	4.15%	6.64%	3.78%	9.25%	-	3.01%
*PG*2	1000001	3.96%	6.52%	3.69%	8.24%	-	2.97%
*PG*3	4000001	3.75%	5.98%	3.36%	8.11%	-	2.52%
*PG*4	9000001	3.25%	5.91%	3.24%	8.02%	-	2.23%

Table 10. r^2 score of the EM aging models tested by different regression techniques For IBM power grid benchmarks.

PG Circuits	#interconnects	r^2 score					
		Bayesian	Random Forest	Ridge	SVM	Gaussian	Neural Network
*PG*1	10001	0.932	0.941	0.963	0.825	0.996	0.994
*ibmpg*1	30027	0.971	0.952	0.971	0.855	0.997	0.995
*ibmpg*2	208325	0.937	0.926	0.935	0.832	-	0.986
*ibmpg*3	1401572	0.920	0.914	0.917	0.815	-	0.975
*ibmpg*4	1560645	0.921	0.935	0.921	0.821	-	0.977
*ibmpg*5	1076848	0.926	0.942	0.925	0.829	-	0.983
*ibmpg*6	1649002	0.929	0.925	0.926	0.828	-	0.981
*ibmpgnew*1	2352355	0.921	0.911	0.922	0.823	-	0.978
*ibmpgnew*2	1422830	0.916	0.922	0.914	0.814	-	0.975
*PG*2	1000001	0.922	0.932	0.951	0.855	-	0.995
*PG*3	4000001	0.936	0.941	0.965	0.857	-	0.997
*PG*4	9000001	0.941	0.945	0.971	0.856	-	0.998



(a)

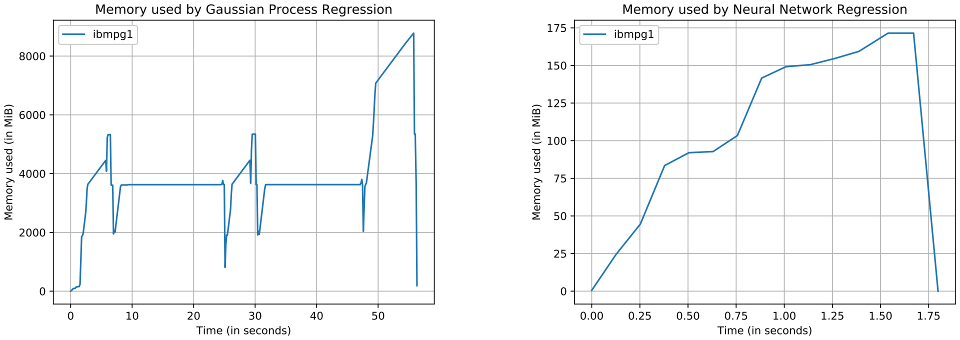
(b)

Fig. 10. Comparison of prediction accuracy on test set in MSE with variations in perturbations size for (a) *ibmpg*2 (b) *ibmpg*4 benchmark circuit.

prediction accuracy on the test set in MSE with variations in perturbations size for *ibmpg*2 and *ibmpg*4 circuit is shown in Fig. 10. It can be understood from Fig. 10 that the MSE of our EM-aging prediction model increases as the size of the perturbation size increases. From this experiment, we want to reiterate the fact that our EM-aging prediction model is best for those incremental designs of the PG network where the perturbations are the least (much less than 10%). We also want to show that as the size of the perturbation increases the EM-aging prediction accuracy decreases. Therefore, our proposed EM-aging prediction model is only applicable to those incremental designs where little perturbations (around 10%) are considered.

5.7 Comparison of Peak Memory and CPU runtime of ML models

In this section, we have shown the memory used by our EM aging model using Neural Network model. We have also done the comparative study of the memory used by EM aging model with different regression techniques and listed the peak memory result for each of the benchmark circuits in Table 11. Although the Gaussian Process Regression (GPR) technique is believed to be the best regression technique due to least MSE value and a r^2 score closer to 1 compared to the other regression technique, as shown in previous section. It is observed that the GPR takes huge memory compared to the other regression models. Due to the huge memory requirement, it has failed to give any result for large benchmark circuits in our machine with 32 GB memory. For *ibmpg*1 circuit memory used during runtime by the EM aging model with GPR and Neural Network technique is shown in Fig. 11. We can observe from the figure that the peak memory used by GPR is more than 8000 Mebibyte (MiB), whereas peak memory used by Neural Network is 174 MiB. The reason behind huge memory taken by the GPR technique is that it stores data in matrices form and then do matrix computation to predict the output feature.



(a)

(b)

Fig. 11. Memory used by EM aging model during MTTF prediction of *ibmpg*1 benchmark circuit using (a) GPR model and (b) Neural Network model. It is to be noted 1 Gigabyte (GB) = 953.674 Mebibyte (MiB)

Similarly, it can be seen that the time taken by the GPR technique is also very high in as shown in Table 12 for the *PG*1, and *ibmpg*1 benchmark circuits. We can also observe that the Bayesian and Ridge regression technique take less time than the Neural Network. However, the Bayesian and Ridge regression techniques don't give accurate results as shown in the Table 9 with higher value of MSE. These two techniques don't fit properly for this EM-aging prediction model compared to the Neural Network which is also known from the r^2 score as listed in Table 9.

Table 11. Peak memory used by EM aging prediction models using different regression techniques For IBM power grid benchmarks.

PG Circuits	**#interconnects**	**Peak memory (in MiB)**					
		Bayesian	**Random Forest**	**Ridge**	**SVM**	**Gaussian**	**Neural Network**
*PG*1	10001	172.75	173.59	172.95	172.21	1146.00	173.09
*ibmpg*1	30027	174.60	174.41	174.80	173.89	8786.51	174.96
*ibmpg*2	208325	205.52	203.90	205.48	280.69	-	205.61
*ibmpg*3	1401572	533.14	506.62	501.26	845.75	-	523.46
*ibmpg*4	1560645	575.96	541.58	540.44	898.79	-	565.51
*ibmpg*5	1076848	444.94	442.87	445.15	733.07	-	487.08
*ibmpg*6	1649002	601.45	557.51	563.92	932.08	-	590.18
*ibmpgnew*1	2352355	792.03	729.89	738.39	1178.69	-	774.69
*ibmpgnew*2	1422830	535.95	502.84	503.63	848.76	-	520.94
*PG*2	1000001	673.81	657.38	628.37	1037.5	-	667.37
*PG*3	4000001	2208.02	2092.046	1964.62	3090.55	-	2208.94
*PG*4	9000001	3594.94	4072.86	3183.58	5333.87	-	3595.95

Table 12. CPU Runtime of the EM aging models tested using different regression techniques For IBM power grid benchmarks.

PG Circuits	**#interconnects**	**CPU Runtime (in Seconds)**					
		Bayesian	**Random Forest**	**Ridge**	**SVM**	**Gaussian**	**Neural Network**
*PG*1	10001	0.307	0.814	0.308	0.322	4.644	0.371
*ibmpg*1	30027	1.106	1.852	1.106	1.165	54.876	1.800
*ibmpg*2	208325	5.240	7.602	5.222	6.249	-	6.304
*ibmpg*3	1401572	10.37	30.052	10.336	32.659	-	31.827
*ibmpg*4	1560645	11.492	35.098	11.418	50.125	-	23.991
*ibmpg*5	1076848	15.739	31.160	15.715	28.565	-	24.196
*ibmpg*6	1649002	21.665	46.253	21.628	61.127	-	34.871
*ibmpgnew*1	2352355	28.711	65.250	28.607	85.260	-	47.64
*ibmpgnew*2	1422830	16.910	35.818	16.879	51.593	-	30.018
*PG*2	1000001	25.623	50.766	25.581	74.736	-	36.243
*PG*3	4000001	101.001	201.612	100.788	357.095	-	151.839
*PG*4	9000001	247.149	478.847	246.717	926.052	-	364.853

5.8 Results of EM-affected Metal Segment Identification

The results of logistic regression-based identification of the EM-affected metals segments are listed in Table 13. Different metrics are mentioned to obtain the prediction quality of the logistic regression-based classification model. From the accuracy metrics, we know about the total accuracy of the model. As the size of the dataset increases the classification accuracy also increase. Therefore, for large power grid benchmark circuits the classification accuracy is higher. The Precision metric is a good measure to determine the number of false positives as mentioned earlier. The Recall metric states the number of actual positives classified by the model. In order to obtain a balance between precision and recall, F1-score is obtained. For the results we have got for classification, almost for all the power grid benchmarks, the metrics obtained are good enough for detecting the mortal interconnects.

6 CONCLUSION AND FUTURE WORK

This paper presents an approach to predict the electromigration (EM)-based aging of the on-chip power grid (PG) network using a machine learning method. Neural Network regression-based machine learning technique is used to predict the mean-time-to-failure (MTTF) during the

Table 13. Results for accuracy of the logistic regression-based EM-affected metal segment identification.

PG Circuits	Classification Accuracy	Precision	Recall	f1-score
*PG*1	0.9701	0.9844	0.9798	0.9821
*ibmpg*1	0.9715	0.9872	0.9781	0.9826
*ibmpg*2	0.9750	0.9868	0.9857	0.9863
*ibmpg*3	0.9800	0.9899	0.9877	0.9888
*ibmpg*4	0.9810	0.9910	0.9879	0.9893
*ibmpg*5	0.9805	0.9895	0.9873	0.9882
*ibmpg*6	0.9865	0.9910	0.9938	0.9924
*ibmpgnew*1	0.9965	0.9972	0.9988	0.9980
*ibmpgnew*2	0.9886	0.9911	0.9939	0.9925
*PG*2	0.9975	0.9983	0.9988	0.9986
*PG*3	0.9983	0.9994	0.9988	0.9991
*PG*4	0.9991	0.9994	0.9994	0.9994

incremental design of the PG network. The training set is generated using the parameters of the PG network, and appropriate features are selected for the proposed machine learning approach by evaluating r^2 score. To speed up the training data generation phase, KLU solver is used for power grid analysis in order to extract the features. For generating the test dataset, a perturbation in the PG network is done, and incremental power grid analysis using KLU solver is used to speed up the process. The trained model is used on the test set for MTTF prediction of different power grid benchmarks. A new failure criterion is proposed in order to improve the MTTF of the proposed model. Results on different power grid benchmark circuits show that the proposed machine learning model exhibits a significant speedup than all of the state-of-the-art EM-based MTTF prediction models. The MTTF predicted by our proposed model is also better than some models and comparable to the most accurate model. Further, we have also proposed a logistic regression-based classification model in order to detect potentially degraded PG interconnects. We also demonstrated different performance and accuracy metrics of the Neural Network model with other standard regression methods in terms of r^2 score, mean-square-error (MSE), CPU runtime, and peak memory used. Neural Network is found to be the best among all the models.

From our work and experimental results, we can recommend the following key points,

- Machine learning-based approach can be applicable for EM aging prediction in incremental PG design.
- Neural network-based supervised machine learning model is the best among all well-known machine learning technique, for the EM aging prediction in incremental PG design in terms of r^2 score, MSE, and peak memory consumption metrics.
- A significant speedup over the state-of-the-arts works can be achieved using the machine learning approach.
- The MTTF value obtained using our machine learning model is most close to the accurate physics-based model reported in the literature, and compared to the other state-of-the-art models.
- Speeding up the MTTF prediction process helps in overall design sign-off time.
- For larger perturbations in the test set, the machine learning technique incurs a significant MSE. As a result, our proposed machine learning model is only applicable for EM aging prediction in incremental PG design.

In the future, robust machine learning models can be explored for EM aging prediction problem, which can produce prediction results with a high confidence score. Further, EM aging prediction using new emerging deep learning techniques can also be explored.

REFERENCES

[1] Online accessed December 2018. The Gold Standard for Accurate Circuit Simulation. (Online accessed December 2018). https://www.synopsys.com/verification/ams-verification/circuit-simulation/hspice.html

[2] Online accessed June 2019. Scikit-learn - Machine Learning in Python. (Online accessed June 2019). https://scikit-learn.org/stable/

[3] Brian Bailey. 2018. Chip Aging Becomes Design Problem. (2018). https://semiengineering.com/chip-aging-becomes-design-problem/

[4] James R Black. 1969. Electromigration: A brief survey and some recent results. *IEEE Transactions on Electron Devices* 16, 4 (1969), 338–347.

[5] Illan A Blech. 1976. Electromigration in thin aluminum films on titanium nitride. *Journal of Applied Physics* 47, 4 (1976), 1203–1208.

[6] Baktash Boghrati and Sachin S Sapatnekar. 2014. Incremental analysis of power grids using backward random walks. *ACM Transactions on Design Automation of Electronic Systems (TODAES)* 19, 3 (2014), 31.

[7] Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)* 2, 3 (2011), 27.

[8] Sandeep Chatterjee, Mohammad Fawaz, and Farid N Najm. 2015. Redundancy-aware power grid electromigration checking under workload uncertainties. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 34, 9 (2015), 1509–1522.

[9] Sandeep Chatterjee, Valeriy Sukharev, and Farid N Najm. 2017. Fast physics-based electromigration assessment by efficient solution of linear time-invariant (LTI) systems. In *Proceedings of the 36th International Conference on Computer-Aided Design*. IEEE Press, 659–666.

[10] Sandeep Chatterjee, Valeriy Sukharev, and Farid N Najm. 2018. Power grid electromigration checking using physics-based models. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 37, 7 (2018), 1317–1330.

[11] Chase Cook, Zeyu Sun, Ertugrul Demircan, Mehul D Shroff, and Sheldon X-D Tan. 2018. Fast electromigration stress evolution analysis for interconnect trees using krylov subspace method. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 26, 5 (2018), 969–980.

[12] Timothy A Davis and Ekanathan Palamadai Natarajan. 2010. Algorithm 907: KLU, a direct sparse solver for circuit simulation problems. *ACM Transactions on Mathematical Software (TOMS)* 37, 3 (2010), 36.

[13] Sukanta Dey, Satyabrata Dash, Sukumar Nandi, and Gaurav Trivedi. 2017. Markov chain model using lévy flight for VLSI power grid analysis. In *2017 30th International Conference on VLSI Design and 2017 16th International Conference on Embedded Systems (VLSID)*. IEEE, 107–112.

[14] Sukanta Dey, Satyabrata Dash, Sukumar Nandi, and Gaurav Trivedi. 2018. PGIREM: Reliability-Constrained IR Drop Minimization and Electromigration Assessment of VLSI Power Grid Networks using Cooperative Coevolution. In *2018 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*. IEEE, 40–45.

[15] Sukanta Dey, Sukanta Nandi, and Gaurav Trivedi. 2020. PowerPlanningDL: Reliability-Aware Framework for On-Chip Power Grid Design using Deep Learning. In *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 1520–1525.

[16] Ibrahim Abe M Elfadel, Duane S Boning, and Xin Li. 2018. Machine Learning in VLSI Computer-Aided Design. (2018).

[17] T. Hastie, R. Tibshirani, and J.H. Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

[18] Tin Kam Ho. 1995. Random decision forests. In *Document analysis and recognition, 1995., proceedings of the third international conference on*, Vol. 1. IEEE, 278–282.

[19] Xin Huang, Armen Kteyan, Sheldon X-D Tan, and Valeriy Sukharev. 2016. Physics-based electromigration models and full-chip assessment for power grid networks. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 35, 11 (2016), 1848–1861.

[20] Yu-Jung Huang, Chung-Long Pan, Shin-Chun Lin, and Mei-Hui Guo. 2018. Machine-Learning Approach in Detection and Classification for Defects in TSV-Based 3-D IC. *IEEE Transactions on Components, Packaging and Manufacturing Technology* 8, 4 (2018), 699–706.

[21] P Inna, Renatas Jakushokas, Mikhail Popovich, Andrey V Mezhiba, Selçuk Köse, Eby G Friedman, et al. 2016. Closed-Form Expressions for Fast IR Drop Analysis. In *On-Chip Power Delivery and Management*. Springer, 373–396.

[22] Andrew B Kahng, Siddhartha Nath, and Tajana S Rosing. 2013. On potential design impacts of electromigration awareness. In *2013 18th Asia and South Pacific Design Automation Conference (ASP-DAC)*. IEEE, 527–532.

[23] MA Korhonen, P Bo/rgesen, KN Tu, and Che-Yu Li. 1993. Stress evolution due to electromigration in confined metal lines. *Journal of Applied Physics* 73, 8 (1993), 3790–3799.

[24] Selçuk Köse and Eby G Friedman. 2012. Efficient algorithms for fast IR drop analysis exploiting locality. *Integration* 45, 2 (2012), 149–161.

[25] Ki-Don Lee. 2012. Electromigration recovery and short lead effect under bipolar-and unipolar-pulse current. In *Reliability Physics Symposium (IRPS), 2012 IEEE International*. IEEE, 6B–3.

[26] Vivek Mishra and Sachin S Sapatnekar. 2016. Predicting electromigration mortality under temperature and product lifetime specifications. In *Proceedings of the 53rd Annual Design Automation Conference*. ACM, 43.

[27] Kevin P Murphy. 2012. *Machine learning: a probabilistic perspective*. Cambridge, MA.

[28] Sarath Mohanachandran Nair, Raiendra Bishnoi, Mehdi B Tahoori, Houman Zahedmanesh, Kristof Croes, Kevin Garello, Gouri Sankar Kar, and Francky Catthoor. 2019. Variation-Aware Physics-Based Electromigration Modeling and Experimental Calibration for VLSI Interconnects. In *2019 IEEE International Reliability Physics Symposium (IRPS)*. IEEE, 1–6.

[29] Farid N Najm. 2015. Physical Design Challenges in the Chip Power Distribution Network. In *Proceedings of the 2015 Symposium on International Symposium on Physical Design*. ACM, 101–101.

[30] Farid N Najm and Valeriy Sukharev. 2019. Efficient Simulation of Electromigration Damage in Large Chip Power Grids Using Accurate Physical Models. In *2019 IEEE International Reliability Physics Symposium (IRPS)*. IEEE, 1–10.

[31] Sani R Nassif. 2008. Power grid analysis benchmarks. In *Proceedings of the 2008 Asia and South Pacific Design Automation Conference*. IEEE Computer Society Press, 376–381.

[32] CS229 Lecture Notes. Online accessed June 2019. Deep Learning. (Online accessed June 2019). http://cs229.stanford.edu/notes/cs229-notes-deep_learning.pdf

[33] Haifeng Qian, Sani R Nassif, and Sachin S Sapatnekar. 2005. Power grid analysis using random walks. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 24, 8 (2005), 1204–1224.

[34] Carl Edward Rasmussen. 2004. Gaussian processes in machine learning. In *Advanced lectures on machine learning*. Springer, 63–71.

[35] Valeriy Sukharev, Xin Huang, Hai-Bao Chen, and Sheldon X-D Tan. 2014. IR-drop based electromigration assessment: parametric failure chip-scale analysis. In *Proceedings of the 2014 IEEE/ACM International Conference on Computer-Aided Design*. IEEE Press, 428–433.

[36] Valeriy Sukharev and Farid N Najm. 2018. Electromigration Check: Where the Design and Reliability Methodologies Meet. *IEEE Transactions on Device and Materials Reliability* 18, 4 (2018), 498–507.

[37] Sheldon X-D Tan, Hussam Amrouch, Taeyoung Kim, Zeyu Sun, Chase Cook, and Joerg Henkel. 2018. Recent advances in EM and BTI induced reliability modeling, analysis and optimization. *Integration* 60 (2018), 132–152.

[38] Xiaoyi Wang, Hongyu Wang, Jian He, Sheldon X-D Tan, Yici Cai, and Shengqi Yang. 2017. Physics-based electromigration modeling and assessment for multi-segment interconnects in power grid networks. In *Proceedings of DATE*. IEEE/ACM, 1731–1736.

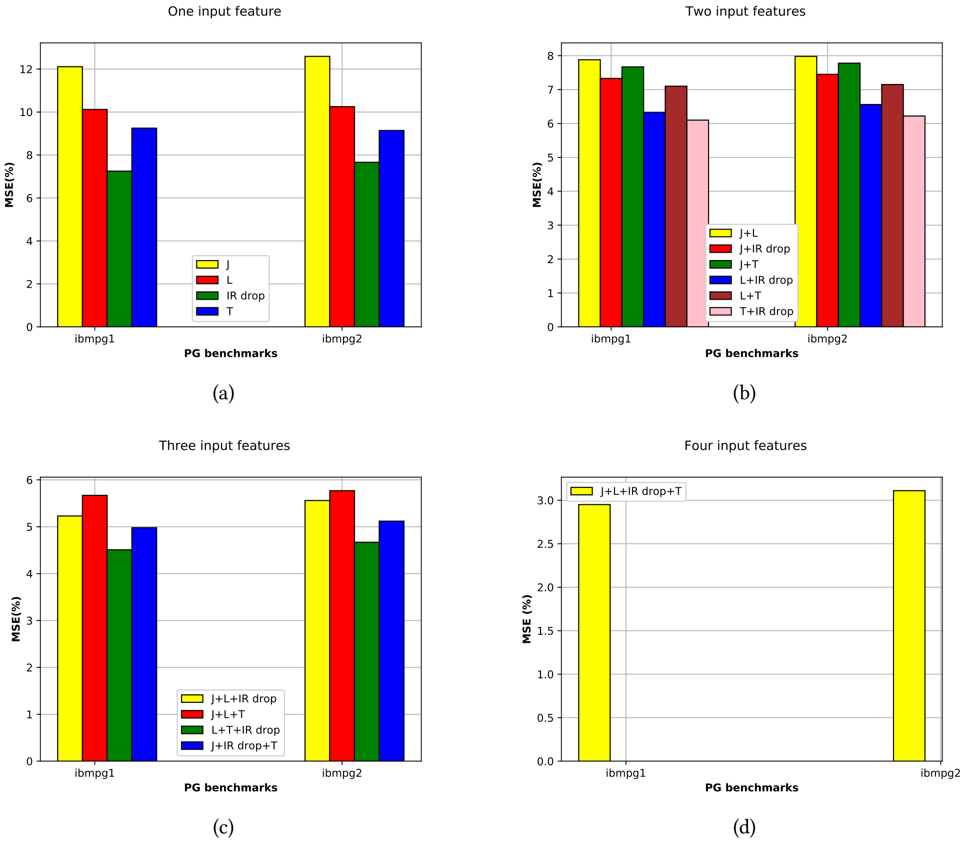
[39] Xiaoyi Wang, Yan Yan, Jian He, Sheldon X-D Tan, Chase Cook, and Shengqi Yang. 2017. Fast physics-based electromigration analysis for multi-branch interconnect trees. In *Proceedings of the 36th International Conference on Computer-Aided Design*. IEEE Press, 169–176.

[40] Bei Yu, Xiaoqing Xu, Subhendu Roy, Yibo Lin, Jiaojiao Ou, and David Z Pan. 2016. Design for manufacturability and reliability in extreme-scaling VLSI. *Science China Information Sciences* 59, 6 (2016), 061406.

[41] Yu Zhong and Martin DF Wong. 2008. Thermal-aware IR drop analysis in large power grid. In *9th International Symposium on Quality Electronic Design (isqed 2008)*. IEEE, 194–199.

APPENDIX: ABLATION STUDY OF DIFFERENT COMBINATIONS OF INPUT FEATURES

To know the model property more intensively, we have performed the ablation study of different combinations of input features. We have obtained the MSE of our Neural Network-based EM aging model for different combinations of input features for all the IBM PG benchmarks, as shown in Fig. 12 and Table 14. From Fig. 12(d), it is clear that for all the four combinations of the feature, the MSE value is less. Therefore, the combination of the four features is suitable for the EM aging model. Further, we have also evaluated the r^2 score for different combinations of input features for all the IBM PG benchmarks, which is listed in Table 15. In this case, also, we have obtained that the models fit best for all the four combinations of input features, as observed from the r^2 score. Therefore, our combinations of input features J, L, IR drop, T and hyperparameters (mentioned in Section 5.1) for the neural network-based EM-aging ML model gives the best result in terms of r^2 score and MSE.



(a) (b) (c) (d)

Fig. 12. Comparison of MSE with different combination of input features for *ibmpg*1 and *ibmpg*2 circuit (a) only one feature (b) combining two features (c) combining three features and (d) combining all four features.

Received June 2019; revised March 2020; accepted May 2020

Table 14. Comparison of MSE with different combination of input features for all the PG benchmarks. All table entries represent MSE value.

	PG Circuits	**J**	**L**	**IR drop**	**T**		
One input features	ibmpg3	12.55%	10.23%	7.56%	9.45%		
	ibmpg4	13.25%	10.87%	7.74%	9.76%		
	ibmpg5	13.10%	10.54%	7.62%	9.59%		
	ibmpg6	11.22%	9.87%	7.12%	9.11%		
	ibmpgnew1	11.45%	9.95%	7.23%	9.10%		
	ibmpgnew2	11.10%	9.93%	7.44%	9.23%		
	PG2	13.12%	10.66%	7.81%	9.89%		
	PG3	10.45%	9.56%	7.25%	9.45%		
	PG4	10.44%	9.44%	7.24%	9.65%		
Two input features	**PG Circuits**	**J+L**	**J+IR drop**	**J+T**	**L+IR drop**	**L+T**	**T+IR drop**
	ibmpg3	7.89%	7.31%	7.70%	6.45%	7.12%	6.10%
	ibmpg4	7.82%	7.26%	7.63%	6.35%	7.11%	6.09%
	ibmpg5	7.75%	7.25%	7.55%	6.28%	7.10%	6.09%
	ibmpg6	7.63%	7.20%	7.52%	6.26%	7.08%	6.10%
	ibmpgnew1	7.55%	7.16%	7.50%	6.20%	7.05%	6.08%
	ibmpgnew2	7.54%	7.16%	7.50%	6.20%	7.01%	6.08%
	PG2	7.74%	7.25%	7.58%	6.35%	7.09%	6.10%
	PG3	7.46%	7.15%	7.48%	6.15%	6.99%	5.98%
	PG4	7.25%	7.10%	7.40%	6.11%	6.95%	5.91%
Three input features	**PG Circuits**	**J+L+IR drop**	**J+L+T**	**L+T+IR drop**	**J+IR drop + T**		
	ibmpg3	5.35%	6.51%	4.55%	5.11%		
	ibmpg4	5.25%	6.49%	4.50%	5.12%		
	ibmpg5	5.23%	6.48%	4.52%	5.10%		
	ibmpg6	5.20%	6.45%	4.53%	5.08%		
	ibmpgnew1	5.15%	6.41%	4.44%	4.99%		
	ibmpgnew2	5.15%	6.41%	4.44%	4.99%		
	PG2	5.21%	6.42%	4.51%	5.05%		
	PG3	5.06%	6.39%	4.39%	4.92%		
	PG4	5.01%	6.35%	4.30%	4.88%		
Four input features	**PG Circuits**	**J+L+IR drop + T**					
	ibmpg3	2.74%					
	ibmpg4	2.89%					
	ibmpg5	3.31%					
	ibmpg6	3.20%					
	ibmpgnew1	3.15%					
	ibmpgnew2	3.01%					
	PG2	2.97%					
	PG3	2.52%					
	PG4	2.23%					

Table 15. Comparison of r^2 score with different combination of input features for all the PG benchmarks. All table entries represent r^2 score.

	PG Circuits	**J**	**L**	**IR drop**	**T**		
One input features	ibmpg1	0.638	0.753	0.859	0.781		
	ibmpg2	0.628	0.742	0.852	0.773		
	ibmpg3	0.645	0.754	0.855	0.775		
	ibmpg4	0.643	0.754	0.854	0.782		
	ibmpg5	0.641	0.750	0.856	0.783		
	ibmpg6	0.653	0.759	0.859	0.784		
	ibmpgnew1	0.652	0.761	0.861	0.783		
	ibmpgnew2	0.655	0.760	0.861	0.784		
	PG2	0.631	0.758	0.857	0.783		
	PG3	0.661	0.763	0.863	0.787		
	PG4	0.663	0.765	0.865	0.788		
Two input features	**PG Circuits**	**J+L**	**J+IR drop**	**J+T**	**L+IR drop**	**L+T**	**T+IR drop**
	ibmpg1	0.763	0.871	0.783	0.884	0.789	0.890
	ibmpg2	0.761	0.869	0.788	0.886	0.790	0.890
	ibmpg3	0.762	0.870	0.786	0.886	0.790	0.891
	ibmpg4	0.754	0.869	0.785	0.884	0.791	0.892
	ibmpg5	0.759	0.870	0.785	0.886	0.792	0.893
	ibmpg6	0.759	0.870	0.786	0.886	0.791	0.894
	ibmpgnew1	0.764	0.871	0.787	0.885	0.791	0.894
	ibmpgnew2	0.764	0.871	0.787	0.884	0.792	0.895
	PG2	0.755	0.867	0.783	0.883	0.788	0.890
	PG3	0.762	0.871	0.788	0.885	0.793	0.896
	PG4	0.764	0.875	0.789	0.887	0.795	0.898
Three input features	**PG Circuits**	**J+L+IR drop**	**J+L+T**	**L+T+IR drop**	**J+IR drop + T**		
	ibmpg1	0.889	0.802	0.914	0.902		
	ibmpg2	0.890	0.803	0.916	0.903		
	ibmpg3	0.891	0.804	0.915	0.903		
	ibmpg4	0.892	0.804	0.915	0.902		
	ibmpg5	0.892	0.805	0.916	0.905		
	ibmpg6	0.893	0.805	0.917	0.907		
	ibmpgnew1	0.894	0.806	0.918	0.907		
	ibmpgnew2	0.894	0.806	0.917	0.906		
	PG2	0.890	0.802	0.915	0.905		
	PG3	0.894	0.807	0.921	0.908		
	PG4	0.895	0.809	0.922	0.909		
Four input features	**PG Circuits**	**J+L+IR drop + T**					
	ibmpg1	0.995					
	ibmpg2	0.986					
	ibmpg3	0.975					
	ibmpg4	0.977					
	ibmpg5	0.983					
	ibmpg6	0.981					
	ibmpgnew1	0.978					
	ibmpgnew2	0.975					
	PG2	0.995					
	PG3	0.997					
	PG4	0.998					