IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Deep Learning-Powered Fault Detection In Digital VLSI Circuits: Advancements And Applications

[1]Anoop S, [2]Ajeesh S, [3]Deebu U S

[1]Assistant professor, [2]Assistant professor, [3]Assistant professor

[1]Electronics and Communication engineering,

[1]College of Engineering, Kottarakkara, Kerala

[2]Computer Science and engineering

[2]College of engineering, Kottarakkara ,Kerala

[3]Electronics and Communication engineering,

[3]College of engineering, Adoor, Kerala

Abstract: Identifying and correcting faults in IC design have become critical stages as the complexity of digital VLSI circuits continues to grow. Presented in this paper is a novel fault identification model based on deep learning (DL), utilizing a distinct type of artificial neural network (ANN) known as stacked sparse autoencoder (SSAE). The main goal of this proposed model is to tackle the challenge in the exploration domain by employing SSAE for identifying features and detecting anomalies in extensive electronic circuits. The model comprises three key stages: test pattern creation, feature reduction, and fault detection. Unsupervised learning using training data is implemented in the SSAE phase to enhance feature extraction. The evaluation of feature extraction effectiveness involves modifying the architecture of the SSAE network. The strategy achieves 99.3% fault coverage with ATALANTA and reduces features by 99.7% using SSAE for test patterns**.**

Index Terms **-** Automatic Test Pattern Generation, ANN, Fault Detection, Digital Circuit, ML, SSAE, Test Pattern.

I. INTRODUCTION

In the rapidly advancing landscape of digital technologies [1], the significance of authentication, error Identification, and correction processes in computer-based systems has reached unprecedented levels. This is particularly true for large and complex digital systems, where fault detection becomes a formidable challenge due to the presence of diverse design bugs and the sheer size of circuits [2]. In response to these challenges, recent research endeavors have focused on enhancing fault detection performance through the application of AI [3]. The complexity of debugging and correction in Integrated Circuit (IC) design stages is underscored by their substantial contribution [4], averaging 70% of the total designs time. Efforts concentrated developing efficient algorithms for debugging and correction to align with behavioral specifications, ultimately reducing time and cost. Specifically, the study explores how ML algorithms, relying on diverse datasets derived from test pattern generators, can effectively detect stuck-at faults. It also investigates the application of SAT-based algorithms [5] for detecting gate replacement errors. The significance of the dataset in ML models is emphasized, with digital Very Large-Scale Integration (VLSI) circuits utilizing outputs from test pattern generators.

Various advanced Automatic Test Pattern Generation tools [6], including the cost-effective ATALANTA software developed at Virginia Tech University, are discussed for their efficiency, flexibility, and fault coverage capabilities. Traditional methods are contrasted with newer approaches that leverage DL techniques, such as ANNs, for improved fault detection and correction processes. SAT and MAX-SAT algorithms are explored as essential tools for detecting and correcting specific errors, such as gate replacements. The conversion of the entire process into Boolean satisfiability problems and the use of advanced SAT solvers are detailed, showcasing their efficacy in detecting specific information and attributes related to fault instances.

The current study proposed an efficient method grounded in DL for Identifying features and detecting faults in combinational digital circuits. The study utilizes stacked sparse autoencoders to reduce big data and extract crucial features with high accuracy. The subsequent sections provide a comprehensive exploration of the background of DL and autoencoders, the proposed model, and a performance evaluation, culminating in a concluding section.

II. LITERATURE REVIEW

Moness et al. [7] presented a groundbreaking semi-supervised Fault Detection (FD) model for combinational and sequential circuits using Deep Sparse Autoencoder. Achieving a notable 187x faster running time than SAT solver-based methods, it outperforms classical ML models, obtaining maximum validation accuracies of 99.93% and 99.95% for combinational and sequential circuits, respectively. Surpassing Radial Basis Function Network (RBFN)-based FD models, it attains a maximum validation accuracy of 97.8%. Limitations include dataset dependencies and the need for broader circuit testing. Shokrolahi et al. [8] introduced a fault detection method for analog circuits using a deep Convolutional Neural Network (CNN) with Power Spectrum Density (PSD) as input images. The proposed PSD-CNN method achieves an impressive 99.8% accuracy, outperforming existing techniques. Limitations may include applicability to specific circuit types and the need for diverse datasets.

Khalil et al. [9] put forward an early transistor fault prediction approach using FFT, PCA, and CNN. Applied to 45nm technology circuits, it achieves an impressive 98.93% accuracy, outperforming state-of-the-art methods. Limitations may include specific circuit applicability. The methodology combines FFT for frequency domain fault signature extraction, PCA for dimension reduction, and CNN for final feature presentation and classification. The proposed method exhibits a high accuracy of 98.93% in fault prediction, surpassing other methods. Hardware implementation on Altera Arria 10 GX FPGA consumes 1.08 W, demonstrating practical feasibility.

Arabi et al. [10] launched a framework to identify and categorize individual parametric defects in circuits using ML algorithms in MATLAB. Utilizing Orcad PSpice for data collection and Monte Carlo analysis for simulation, the method achieves remarkable accuracy: 100% for the first circuit, 99.77% for the second, and 99.72% for the third. Limitations may include dependence on simulation accuracy and specific fault types. Overall, the proposed classification algorithm demonstrates superior efficiency, outperforming other research works in fault detection and classification.

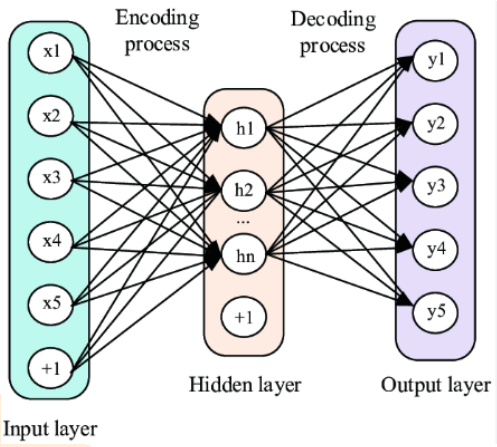
Hussein et al. [11] have done a study for introducing a semi-supervised Fault Detection (FD) model using DL for combinational and sequential circuits. Achieving 99.93% and 99.95% validation accuracy for combinational and sequential circuits, respectively, the model outperforms classical ML models and SAT solver-based methods, showcasing the efficiency of leveraging DL for fault detection in digital circuits. Gaber et al. [12] addressed the impact of digital VLSI circuit complexity on verification methodologies. It proposes an incremental correction algorithm to generate compact test patterns, reducing dependence on specialized patterns. Limitations may include specific circuit dependencies.

Radhakrishnan et al. [13] proposed Efficient Partitioning and Placement-based Fault TSV detection in 3D-IC, addressing challenges in fault detection within Through Silicon Vias (TSVs). Limitations may include specific conditions for optimal performance. Wang et al. [14] introduced an LSTM model for efficient online detection of intermittent faults in electronic systems. The model exhibits high precision (97%) and recall (87%), showcasing its effectiveness. Limitations may include specific conditions for optimal performance.

III. METHODOLOGY

3.1. Autoencoders

An autoencoder (AE), a potent unsupervised ML algorithm, facilitates dimensional reducing characteristics from elevated to diminished dimensions. Comprising encoder, code, and decoder components, it down-samples input to a latent space, preserving vital features while discarding redundant ones. The decoder reconstructs input from the encoding, enabling effective use of encoding data in supervised algorithms like classifiers. Figure 1 depicts a representation of a AE process.



IV. V. **Fig. 1:** Autoencoder process [15]

3.2. Sparse autoencoders

Sparse autoencoders (SAE) aim to maintain hidden layer node count by activating only a few neurons, serving as an alternative to introducing a bottleneck without reduction. This is achieved through a sparsity constraint and penalizing hidden layer activation in the loss function. Figure 2 depicts a representation of a sparse autoencoder.

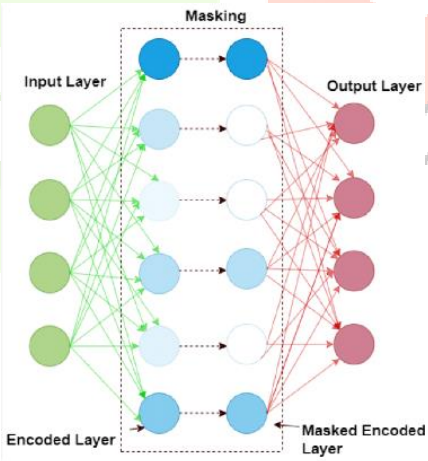
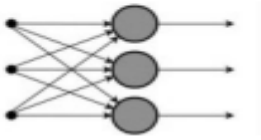
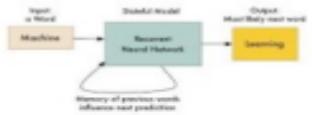
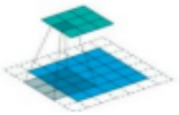
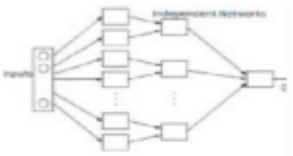


Fig. 2: Sparse autoencoder [16]

3.3. Deep learning

Utilizing DL represents a cutting-edge and effective approach in the realm of automated feature extraction and fault detection. Leveraging large datasets and employing DL to extract crucial features results in enhanced fault detection performance compared to conventional methods. A comparison between various types of neural networks (NN) is depicted in Table 1.

Table 1: Comparison of NNs

Classification of deep NN	Description	Applications	Visual representation
Feedforward NN	- Basic configurations of artificial neural networks (ANN). - Input data flows unidirectionally from input nodes to output nodes. - It may include hidden layers or not. - Backpropagation is absent, often utilizing a classifying activation function. - Operates with a front-propagated wave.	Categorization involving intricate target classes.	
Recurrent NN	- Records the output and loops it back to the input for accurate predictions. - Each neuron functions akin to a memory cell. - The learning rate is employed to determine the accurate prediction.	- Models for converting text to speech (TTS).	
Convolution NN	- Resembles a feedforward neural network. - Input features are processed in batches.	Applicable in the realm of image processing and computer vision.	
Modular NN	- Comprising several independent networks. - Inputs for each neural network differ, facilitating distinct sub-tasks without interaction. - Breaking down extensive processes into smaller models to reduce complexity. -Processing time is contingent on the quantity of neurons.	- Numerous applications like approximating functions, recognizing characters, and developing a patient -independent ECG recognition system.	

3.4. SSAE

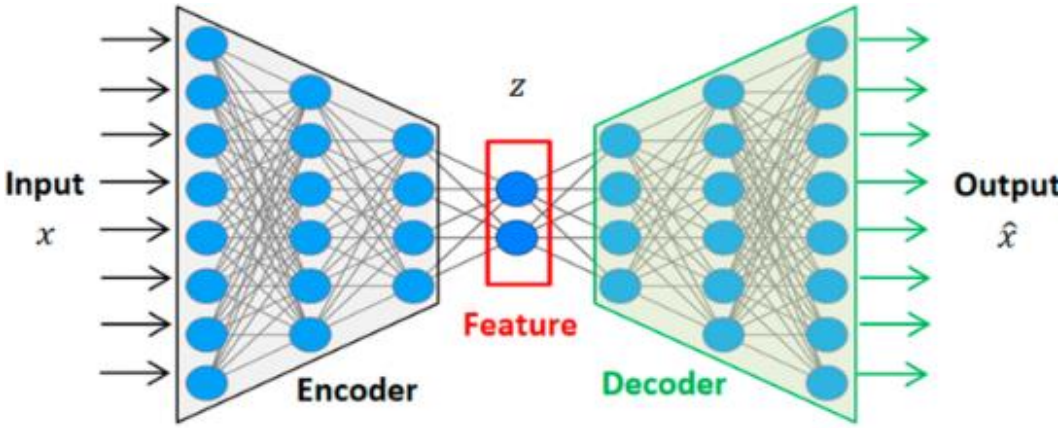


Fig 3: Deep AE [17]

In complex architectures, SSAE or Deep AE enhances the efficiency typical of AE. Comprising multiple stacked layers, each encoder layer becomes the input for the subsequent autoencoder. After separate unsupervised training, the network proceeds to a supervised stage. Figure 3 depicts a representation of a deep autoencoder.

3.5. Fault Detection using DL

Utilizing Stacked SSAE for fault detection in digital VLSI circuits eliminates the need for an accurate mathematical model or formal specifications. Test vectors generated by the powerful ATPG tool ATALANTA serve as inputs for SSAE, engaging unsupervised learning in compressing data. Final softmax layer aid in classification using the softmax function, offering a normalized probability distribution for intuitive analysis.

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{x_j}} \quad (4)$$

z-Input vector of softmax, z_i-output vector of neural network, k-Quantity of categories in the multi-category classifier.

3.6. Suggested Anomaly Detection Methodology

We elaborate on detailed explanation of our DL -based fault detection model, as illustrated in Figure 8. The algorithm comprises three primary stages:

Step 1: Test Pattern Generation

The initial step involves generating unique and extensive test patterns for the digital circuit. Utilizing the ATALANTA tool, based on the FAN algorithm, producing multiple evaluated sequences with accurate results and anomalies masks corresponding to each pattern.

Step 2: Feature Reduction

Post data generation, the neural network uses SSAE for unsupervised learning, crucially reducing feature dimensions in large digital circuits. The goal is minimal features without compromising accuracy, achieved by training on test patterns and fault-free responses. SSAE efficiently distills pertinent features, enhancing overall performance, particularly in fault detection.

Step 3: Fault Detection

In the final stage, faults for each test pattern are identified using a softmax classifier. The classifier transforms real values into probabilities, aiding precise fault detection and enhancing overall effectiveness.

VI. RESULT AND DISCUSSION

The algorithm employs a SSAE for detecting stuck-at-0 and stuck-at-1 faults and applied to 8 integrated circuits, the methodology is derived from the ISCAS'85 benchmark. On an Intel Core i7 10750 processor running at 2.6 GHz with 16 GB system memory, the execution took place. Each ISCAS'85 was processed in the ATALANTA software, identifying stuck-at-0 and stuck-at-1 faults and generating a specific number of test patterns for each fault. For each digital circuit, we selected 30/50 test vectors for each error identification. Table 2 provides information on the faults, the percentage of fault coverage, test vectors and number of inputs and outputs for each electronic circuit.

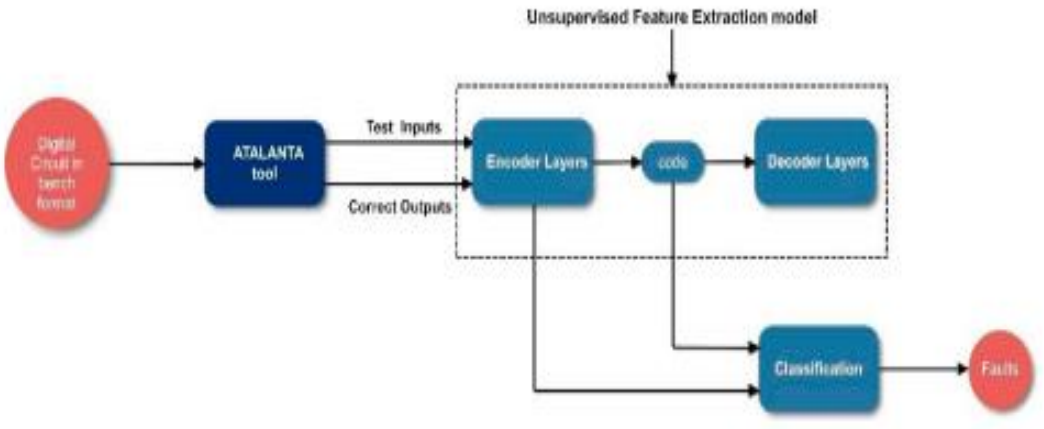


Fig 4: proposed model

Table 2: Fault coverage, fault occurrences, test patterns, and Inputs and outputs quantity for eight combinational circuits.

CNF Type	Inputs and outputs quantity	fault occurrences	test patterns	Fault Coverage
c17	7	22	54	100%
c432	43	524	8950	98.8%
c499	73	758	13066	95.5%
c880	86	942	45356	100%
c1908	58	1879	34216	99.2%
c3540	72	3428	55752	95.9%
c5315	301	5350	88978	98.7%
c6288	64	7744	114101	80.8%

Table 3 outlines SSAE the parameters. Through experimental tuning, adjustments were made to optimize the reduction of features. Additionally, the sparsity constraint (□) underwent experimental fine-tuning to improve the reconstruction performance of SSAE, with considerations for 100 epochs and a batch size of 32. The table also presents a comparison of validation accuracy between simple AE and SSAE. This comparison underscores that SSAE attains a peak reconstruction accuracy of approximately 99.7% for the "c5315" circuit, initially featuring 300 inputs and outputs but compressed to 21 through three SSAE with a sparsity of around 10e-9.

Table 3: The number of concealed neurons and the validation accuracy in feature extraction with sparsity constraints.

CNF Type	Quantity of hidden neurons	Validation Accuracy using simple AE	Validation Accuracy using SSAE	Sparsity Constraints of SSAE
c17	5,3,2	63.9%	71.4%	10e-3
c432	30,20,10	95.4%	97.5%	10e-6
c499	50,30,20	98.2%	98.50%	10.e-9
c880	50,30,20	98.3%	98.6%	10e
c1908	50,30,20	97.8%	98.8%	10e
c3540	50,30,20	97.3%	98.2%	10e
c5315	100,50,20	98.9%	99.7%	10.e-9
c6288	50,30,20	97.9%	99.3%	10e-6

Figure 9 illustrates the impact of sparsity constraint (ρ) on SSAE's accuracy. Experiment involved varying sparse constraint ($\rho\rho_1$=0.50, $\rho\rho_2$=0.067, $\rho\rho_3$=0.024) across five different combinational circuits. The SSAE model implementation employed three Sparse Autoencoders (SAE) with the architecture detailed in Table 3. The results indicate that the optimal sparsity constraint value is approximately 0.024.

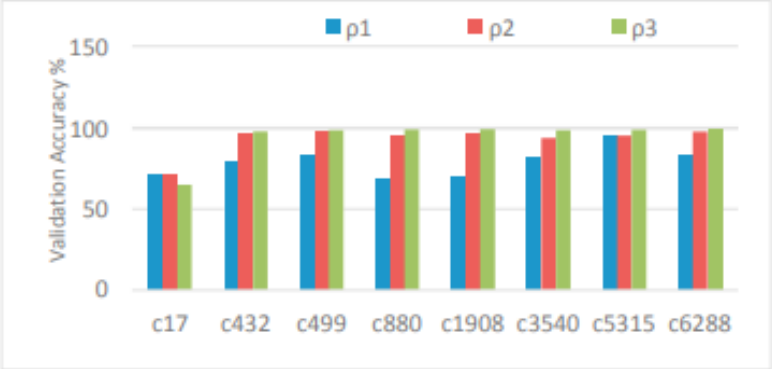


Fig.9: The impact of imposing sparsity constraints on the precision of SSAE.

VII. CONCLUSION

The study presented an innovative ANN-based method for detecting stuck-at-faults in the 27-channel interrupt controller and Arithmetic Logic Units circuits sourced from the ISCAS'85 benchmarks. The algorithm addresses the challenge of search space explosion by compacting digital circuit features through the utilization of test patterns generated by the ATALANTA software. Employing a stack of three sparse AEs enhances the algorithm's capacity to achieve optimal reconstruction accuracy, particularly beneficial for troubleshooting extensive-scale electronic circuits. SSAE is integrated by a softmax classifier for supervised learning, leveraging fault masks. Executed on 8 logical circuits, the approach attains remarkable fault coverage of approximately 99.3% with ATALANTA, while simultaneously achieving around 99.7% using test patterns.

REFERENCES

[1] Orishev, J., & Achilov, S. (2023). DIGITAL TECHNOLOGIES AS AN EDUCATIONAL PROCESS IN PREPARING FUTURE TEACHERS FOR PROJECT ACTIVITIES. Science and innovation, 2(B3), 425-429.

[2] Piliposyan, G. (2023). Investigation into Detection of Hardware Trojans on Printed Circuit Boards (Doctoral dissertation, The University of Liverpool (United Kingdom)).

[3] Abubakar, A., Jibril, M. M., Almeida, C. F., Gemignani, M., Yahya, M. N., & Abba, S. I. (2023). A Novel Hybrid Optimization Approach for Fault Detection in Photovoltaic Arrays and Inverters Using AI and Statistical Learning Techniques: A Focus on Sustainable Environment. Processes, 11(9), 2549.

[4] Milner, L. E. (2023). Unified design flow for SiGe and group III-V broadband microwave integrated circuits utilising custom pcells (Doctoral dissertation, Macquarie University).

[5] Kuhlmann, I., Gessler, A., Laszlo, V., & Thimm, M. (2023). Comparison of SAT-based and ASP-based Algorithms for Inconsistency Measurement. arXiv preprint arXiv:2304.14832.

[6] Zhong, Y., & Guin, U. (2023). A Comprehensive Test Pattern Generation Approach Exploiting the SAT Attack for Logic Locking. IEEE Transactions on Computers.

[7] Moness, M., Gaber, L., Hussein, A. I., & Ali, H. M. (2022). Automated Design Error Debugging of Digital VLSI Circuits. Journal of Electronic Testing, 38(4), 395-417.

[8] Shokrolahi, S. M., & Karimiziarani, M. (2021). A deep network solution for intelligent fault detection in analog circuit. Analog Integrated Circuits and Signal Processing, 107, 597-604.

[9] Khalil, K., Eldash, O., Kumar, A., & Bayoumi, M. (2020). Machine learning-based approach for hardware faults prediction. IEEE Transactions on Circuits and Systems I: Regular Papers, 67(11), 3880-3892.

[10] Arabi, A., Ayad, M., Bourouba, N., Benziane, M., Griche, I., Ghoneim, S. S., ... & Ghaly, R. N. (2023). An efficient method for faults diagnosis in analog circuits based on machine learning classifiers. Alexandria Engineering Journal, 77, 109-125.

[11] Hussein, A., Hanafy, M. A., Mohammed, M., & Lamya, G. (2023). Automated Design Error Debugging of Digital VLSI Circuits.

[12] Gaber, L., Hussein, A. I., & Moness, M. (2020, November). Incremental Automatic Correction for Digital VLSI Circuits. In Proceeeding 11th International Conference on VLSI (VLSI 2020).

[13] Radhakrishnan Nair, R. K., Pothiraj, S., Radhakrishnan Nair, T. R., & Cengiz, K. (2021). An efficient partitioning and placement based fault TSV detection in 3D-IC using deep learning approach. Journal of Ambient Intelligence and Humanized Computing, 1-14.

[14] Wang, X., Jiang, L., & Chakrabarty, K. (2020, April). LSTM-based analysis of temporally- and spatially-correlated signatures for intermittent fault detection. In 2020 IEEE 38th VLSI Test Symposium (VTS) (pp. 1-6). IEEE.

[15] Wang, K., Forbes, M. G., Gopaluni, B., Chen, J., & Song, Z. (2019). Systematic development of a new variational autoencoder model based on uncertain data for monitoring nonlinear processes. IEEE Access, 7, 22554-22565.

[16] Rao, K. N., Rao, K. V., & PVGD, P. R. (2021). A hybrid intrusion detection system based on sparse autoencoder and deep neural network. Computer Communications, 180, 77-88.

[17] Chang, Y., Tu, Z., Xie, W., & Yuan, J. (2020). Clustering driven deep autoencoder for video anomaly detection. In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16 (pp. 329-345). Springer International Pu