International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7632
***Article Received:** 04 June 2023 **Revised:** 26 July 2023 **Accepted:** 14 August 2023*

Machine Learning Techniques to Evaluate the Approximation of Utilization Power in Circuits

Varghese M P[1], T Muthumanickam[2]
[1]Research Scholar, Department of Electronics and Communication Engineering,
Vinayaka Mission's Research Foundation Deemed to be University,
Salem, India
jaishriresch@gmail.com
[2]Professor & Head, Department of Electronics and Communication Engineering,
Vinayaka Mission's Kirupananda Variyar Engineering College,
Vinayaka Mission's Research Foundation Deemed to be University,
Salem, India

Abstract. The need for products that are more streamlined, more useful, and have longer battery lives is rising in today's culture. More components are being integrated onto smaller, more complex chips in order to do this. The outcome is higher total power consumption as a result of increased power dissipation brought on by dynamic and static currents in integrated circuits (ICs). For effective power planning and the precise application of power pads and strips by floor plan engineers, estimating power dissipation at an early stage is essential. With more information about the design attributes, power estimation accuracy increases. For a variety of applications, including function approximation, regularization, noisy interpolation, classification, and density estimation, they offer a coherent framework. RBFNN training is also quicker than training multi-layer perceptron networks. RBFNN learning typically comprises of a linear supervised phase for computing weights, followed by an unsupervised phase for determining the centers and widths of the Gaussian basis functions. This study investigates several learning techniques for estimating the synaptic weights, widths, and centers of RBFNNs. In this study, RBF networks—a traditional family of supervised learning algorithms—are examined. Using centers found using k-means clustering and the square norm of the network coefficients, respectively, two popular regularization techniques are examined. It is demonstrated that each of these RBF techniques are capable of being rewritten as data-dependent kernels. Due to their adaptability and quicker training time when compared to multi-layer perceptron networks, RBFNNs present a compelling option to conventional neural network models. Along with experimental data, the research offers a theoretical analysis of these techniques, indicating competitive performance and a few advantages over traditional kernel techniques in terms of adaptability (ability to take into account unlabeled data) and computing complexity. The research also discusses current achievements in using soft k-means features for image identification and other tasks.

Keywords: EDA, Machine learning, MLP, Neural network , RBFNN.

I. Introduction

Artificial neural networks are created to mimic the organic neural systems' processing capacities. They are made up of numerous interconnected, basic processing units called neurons that are capable of concurrent operation. By modifying the strength of the connections between neurons in response to input data, these networks can learn to solve challenging tasks. They can process information that is noisy, inconsistent, hazy, or uncertain and can learn to adapt to new surroundings. Artificial neural networks have undergone a great deal of research and development as a result of these properties [1]. Power planning is essential for ensuring even power distribution throughout the design in the area of integrated circuit (IC) design. Design engineers can distribute power manually or automatically using backend electronic design automation (EDA) techniques. To carry power supply (VDD and GND) at various levels within the IC, rings and stripes are created [2]. To decrease time to market and design complexity, it has become crucial to estimate power early in the specification phase. It assists floor plan engineers in carrying out accurate power planning and helps in accurately calculating the amount of power strips and pads.

A unique method that necessitates prior knowledge of circuit power dissipation is the use of machine learning (ML) to anticipate power consumption [3]. Power estimation can be trained on a variety of ML models, including support vector machines (SVM), k-nearest neighbors (KNN), random forests (RF), etc. Even these modern technologies produce reliable data, they could take a long time to process. The goal is to help floor plan engineers by providing ML-based power estimation for VLSI circuits to improve power planning [4]. Parallel processing architectures and the capacity to learn from input data are features of artificial neural networks. They can carry out particular jobs by modifying the connection weights through training with data. Iterative neural network training uses a variety of learning algorithms, such as supervised and unsupervised learning [5].

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7632
***Article Received:** 04 June 2023 **Revised:** 26 July 2023 **Accepted:** 14 August 2023*

Given that they are made up of connected, straightforward processing pieces, neural networks have demonstrated success in solving a variety of issues [6]. To understand the nonlinear mapping between input sensor data and output classes is what they are there for. Radial basis function neural networks are a prominent and well-known variety of neural network noted for their outstanding performance. Powell first proposed RBFNNs as a solution to the interpolation issue, where the number of centers needed to precisely match the data points. RBFNNs became more useful for applications requiring a large number of samples after Lowe loosened this constraint and allowed for fewer centers than data samples [7]. The existence of a quick, linear learning process that can express intricate nonlinear mappings is a noteworthy characteristic of RBFNNs. The generalization capabilities of RBFNNs have also received significant attention in the literature. RBFNNs are currently being researched in both the numerical analysis and machine learning fields. The idea of biological receptive fields gave rise to the idea of RBFNNs, and it has been demonstrated that RBFNNs may approximate any nonlinear mapping between stimuli and responses.

With the aid of artificial intelligence (AI), computers are now capable of learning and resolving complex issues. The goal of machine learning, a branch of artificial intelligence, is to simulate the workings of the human brain in computers. Machine learning algorithms are used in clinical trials to simulate the expertise of medical professionals in disease detection and analysis [8-10]. The classification technique is not considerably impacted by flaws in medical databases. Classification is one of the most widely used machine learning methods in medical diagnostics, and Artificial Neural Networks (ANNs) are frequently used [11].

An technique that is often employed combines the Back Propagation (BP) algorithm with the Multilayer Perceptron (MLP) ANN. However, the steepest descent method in the BP algorithm and the nonlinearity of the MLP do not always converge to a globally optimum set of parameters. To choose the best parameter set from a set of locally optimal parameters, it is frequently necessary to do several iterations and use trial and error. The RBF network, which uses linear parameters and has applications in areas like electrical and electronic engineering, is an alternative to MLP [12]. Theoretically, it has been demonstrated that RBF networks can learn without running into local minima, ensuring convergence to optimal parameters.

The literature has presented a variety of regularization techniques with varying theoretical attributes, computational complexity, and actual performance. Kernel machines, such as kernel Support Vector Machines (K-SVM) or kernel regularized least squares (K-RLS) algorithms, are a popular method for regularizing RBF networks [13]. These methods use the norm in a Reproducing Kernel Hilbert Space (RKHS) to restrict the function space. When combined with a radial kernel, kernel methods—which are frequently thought of as a separate class of algorithms—become different forms of RBF networks.

II. Methodology

In this work, we revisit two approaches for training Radial Basis Function (RBF) networks that were initially developed before kernel machines became widely employed in machine learning. We concentrate on regularization using the squared norm of an RBF network's coefficients and center selection using k-means clustering. It's interesting that we may recast these algorithms as kernel techniques that explicitly depend on the data distribution [14]. We point out a number of benefits of these methods over traditional kernel methods, especially in terms of flexibility (such adding unlabeled data) and their capacity to handle huge datasets.

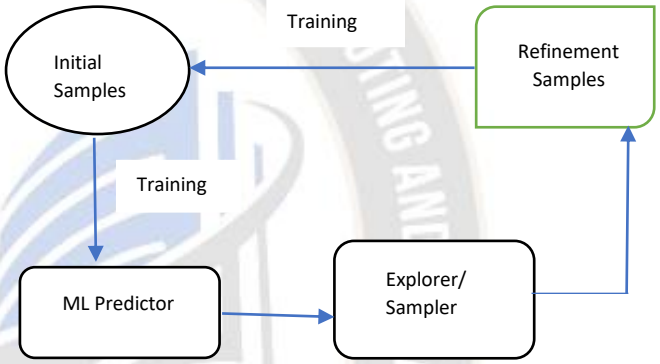

Fig.1 Classification of samples for training and testing

Fig. 1 shows the sampling of the data into training and testing data based on the ML preceptor support. A traditional family of methods for supervised learning is called radial basis function (RBF) networks. By utilizing a linear combination of radial kernels, such as the Gaussian kernel (typically viewed as a two-layer neural network), they aim to approximate the desired function. The centers and weights for these functions are typically the output of an RBF network learning method [15-18]. RBF networks have drawn a lot of interest from the machine learning field because of their excellent performance and advantageous theoretical characteristics. A key component of every RBF network method is capacity control. By penalizing coefficients and choosing a smaller set of centers than the input data, over fitting, in which every data point becomes a center, must be avoided. A well-known and often used type of neural network is the multilayer perceptron (MLP), sometimes known as multi-layered feed-forward neural networks [29,30]. It has many layers, input and output terminals, one or more hidden layers with connected nodes, and input and output terminals. It is advised to adopt a one-step iterative prediction methodology in order to get around

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7632
***Article Received:** 04 June 2023 **Revised:** 26 July 2023 **Accepted:** 14 August 2023*

the drawbacks of a typical one-step prediction method for chaotic time series.

III. RBF neural network architecture:

The theory of function approximation serves as the foundation for the idea behind radial basis function neural networks (RBFNNs). In RBFNNs, the Euclidean distance between each neuron's center and the input point is calculated, and the distance is then subjected to a radial basis function (RBF) to determine the weight or impact of each neuron. The RBF, sometimes referred to as a kernel function or Gaussian function, gets its name from the fact that it uses distance as its argument. RBFs function essentially as local receptors, with the output based on the input's proximity to a stored vector.

An RBFNN's input layer may contain a number of predictor variables, each of which is linked to a separate neuron. The input layer's function is to transmit input vectors to the hidden layer. A bias term and numerous RBF units with Gaussian kernels make up the hidden layer [19-22]. The center and width of the Gaussian function define it. The RBFNN classifier compares the Euclidean distance of the input vector from the Gaussian function's center and then applies a nonlinear change in the hidden layer. When expressed as a matrix, the multi-quadric function, a sort of radial basis function, exhibits a noteworthy spectral property: it is almost negative definite. In two dimensions, Franke found that this kind of basis function offers extremely precise interpolation surfaces [21]. He also discovered that even with a limited number of centers, the inverse multi-quadric basis function can produce excellent approximations. For multi-quadric basis functions, an ideal value for sigma, however, is not yet understood.

When opposed to the Gaussian function, the thin plate spline basis function demonstrates a more global nature, which means that even a slight change in one control point might have an impact on all of the other points' coefficients. Similar to this, certain applications take into account polynomial basis functions like cubic and linear [22-26]. According to research on neural networks, the Gaussian RBF is frequently utilized in a variety of fields, including engineering, control systems, medical and biological science, and more [27]. RBF networks have the advantage of allowing one to select appropriate hidden unit or basis function parameters without having to perform a full network nonlinear optimization. Unsupervised techniques published in the literature can be used to determine the coordinates of each hidden layer's center in the RBF function.

IV. RBFNN training algorithms:

Radial basis functions are a particular kind of artificial neural network that are used as activation functions; these networks are known as RBFNNs.

RBFNNs are trained by changing the network's parameters to reduce the difference between the desired and anticipated results. RBFNNs can be trained using a variety of algorithms, including the following methods:

Gradient Descent: RBFNNs and other neural networks are frequently trained using the gradient descent optimization algorithm. It works by repeatedly computing the gradients of the loss function relative to the weights and updating the weights in the direction opposite to the gradient. RBFNNs can be trained using a variety of gradient descent techniques, including stochastic gradient descent (SGD) and batch gradient descent.

Evolutionary methods: RBFNNs can also be trained using evolutionary methods, such as genetic algorithms or particle swarm optimization. A population of potential solutions is used in these algorithms, and they change over time. Based on their performance on the training data, the population's members are assessed for their fitness.

Fig.2 shows the flow of data and classification from starting to end. The data is processed by using Radial Basis Function Neural network and the error is calculated for evaluation. Based on the error value obtained, the model will be adjusted to improve the efficiency of the model. Clustering-based These techniques include a two-step approach. The first step is clustering the input data using techniques like k-means or Gaussian mixture models [28]. The radial basis functions in the RBFNN are then used as centers at the cluster centers. Following that, algorithms like least squares or gradient descent are used to determine the weights connecting the hidden layer to the output layer. Orthogonal Least Squares (OLS): OLS is a powerful algorithm created specifically for RBFNN training. It starts by choosing a small subset of candidate functions from a wider pool of RBFs. The RBFNN is built using these selected RBFs, and the weights are modified using the least squares technique. Up until the target level of accuracy is reached, the algorithm iteratively adds new RBFs and updates the weights.

RBFNNs are trained using the online learning technique recursive least squares (RLS). As new training samples are supplied, it sequentially updates the weights of the network. Each sample's output is calculated by RLS, which then compares it to the desired output and makes weight adjustments based on the difference between the two. When data centers in an unstructured format, this technique is appropriate. manner. In order to reach the necessary approximation or classification accuracy, different parameters must be determined during the training process of RBFNNs. The number of neurons in the hidden layer (which should ideally be significantly lower than the amount of data points), the hidden node center coordinates, the width (spread) of each RBF, and the weights given to the RBF outputs in the output

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7632
Article Received: 04 June 2023 Revised: 26 July 2023 Accepted: 14 August 2023

layer are some of these factors. These parameters are determined by the training algorithm.

A well-known and often used type of neural network is the multilayer perceptron (MLP), sometimes known as multi-layered feed-forward neural networks [29,30]. It has many layers, input and output terminals, one or more hidden layers with connected nodes, and input and output terminals.

It is advised to adopt a one-step iterative prediction methodology in order to get around the drawbacks of a typical one-step prediction method for chaotic time series.

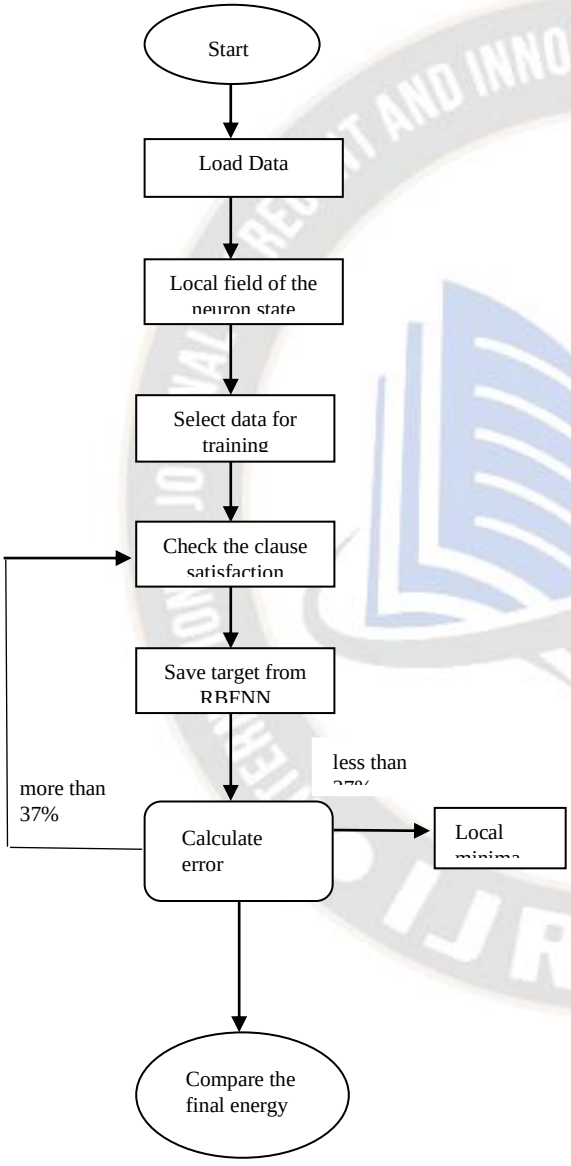

Fig.2 Flow of the proposed work

Radial Basis Function (RBF)-based one-step iterative prediction can be explained as follows: The first step is to normalize the initially erratic time series. Apply the G-P algorithm theory to the RBF network to determine the number of input nodes for each subnet. Based on the one-step prediction value of each subnet, calculate the weighted factor ().Calculate the RBF network's one-step prediction value. Use the Step 4 anticipated result as the next input data for a one-step prediction that follows. Repeat Steps 3 and 4 to gradually obtain the future trend of the actual case (such as Lorenz's attractor or hydraulic pump). The number of iterations in this loop depends on the length of the desired data to be predicted.

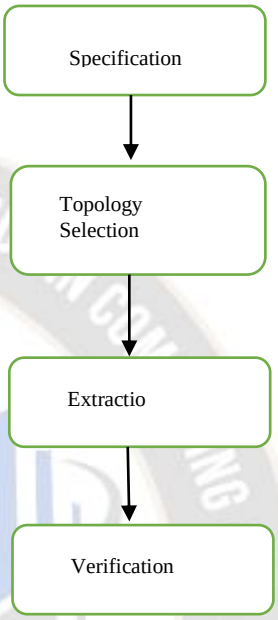

Fig.3. Learning based feature extraction

Fig.3 shows a design flow for analogue circuit design that combines top-down and bottom-up methods. Designers choose an appropriate topology that complies with the system requirements at the circuit level first in the top-down flow. The sizes of certain devices are then optimized at the device level. The pre-layout design phase is made up of the topology design and device sizing steps. The circuit layout is made once the schematic has been carefully designed. The circuit is then modeled using the parasitic effects, a process known as post-layout simulations, after parasitic elements have been removed from the layout. Designers must change the parameters and restart the process if the post-layout simulations do not meet the requirements.

Until the desired layout is obtained, this iterative process may continue. Despite notable improvements in analogue design automation, physical labor continues to be an essential part of the design process. To solve analogue design issues, including topology selection, device scaling, and physical layout creation, researchers are investigating the incorporation of machine learning approaches. The RBFNN training algorithm selected will rely on the unique requirements, dataset size, and computational resources available. Every algorithm has advantages and disadvantages, therefore choosing the best strategy for a particular application frequently requires testing.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7632
***Article Received:** 04 June 2023 **Revised:** 26 July 2023 **Accepted:** 14 August 2023*

V. Selection of RBF Centers by Clustering:

In data analysis, clustering is a useful tool for understanding how a dataset is distributed, especially for locating the centers of Radial Basis Functions (RBFs). The prototypes of these clusters can be used as the centers of the RBFs by clustering the training data. Depending on the clustering algorithm selected, the number of clusters can be predetermined or calculated automatically. The efficiency of learning in RBF networks is substantially influenced by the clustering algorithm's performance.

To cluster RBF centers, unsupervised clustering techniques like C-means are frequently used. However, because they also take into account the distribution of the output patterns, RBF centers identified through supervised clustering typically prove to be more effective for RBF network learning. The LVQ1 technique is frequently used to cluster the RBF centers for classification tasks in RBF network training. Numerous research publications examine the use of clustering for choosing RBF centers, and both supervised and unsupervised clustering algorithms can be used to group RBF centers.

The covariance matrices of the RBFs are then set to coincide with the covariances of the input patterns inside each cluster after the RBF centers have been established. By applying the Mahala Nobis distance, which is determined by the weighted norm, this transforms the Gaussian RBF network into a generalized RBF network.

A crucial and popular method is K-nearest neighbor (KNN), particularly for tasks like categorization and missing data imputation. KNN is used in this situation to impute missing characteristics by replacing them with the most reasonable alternatives. Although there are other versions that can be utilized for imputation, KNN was selected for this investigation because to its usefulness. Euclidean, Makowski, or Manhattan distances are only a few examples of proximity computations used by the straightforward and supervised machine learning algorithm KNN. The following steps make up KNN's workflow:

Along with their accompanying class labels, the distance between each labelled instance and the test instance is calculated. The k closest examples are chosen as the uncertain test instance's "neighbors" based on how similar their class labels are. To identify the class for the unlabeled instance, a survey is given to the chosen neighbors.

VI. Results and Discussions

A larger number of neurons are needed for an RBF network's training process, which generally produces better results with a bigger number of training vectors. The RBF network doesn't respond to the full input space simultaneously, in contrast to other networks. Instead, it determines the input layer's center first and then activates the inputs that are near it. Consequently, the RBF network reacts to local inputs quickly. The radial basis layer and the linear output layer are the two layers that make up the network. Competitive learning, or clustering, is carried out during the training process. To enhance network performance, network parameters like the spread number and goal number are modified [31]. The vector product is created by multiplying the estimated distance between the input and weight vectors by the bias values. Multiple neurons are produced in the input layer as a result of this process, and the output layer then generates the output values.

The UCI Repository's Hepatitis database is selected for classification purposes. The dataset consists of 155 cases with 17 variables, 123 of which are labelled "live," and 32 of which are labelled "die" [32]. The gradient descent-based MLP approach using the Back propagation learning algorithm successfully categorises the data with an accuracy of 80%. However, the classification accuracy increases to 85.80% when the RBF network is used.

Method	Accuracy Rate
RBF Network	86.7%
MLP Network	81.3%

Table 1. Classification Accuracy

Table 1 displays the efficiency of obtained while using the RBF neural network and it is compared with MLP models evaluation. The three research that use machine learning (ML) methods to improve traditional optimization algorithms are summarized in this section. STAGE [33], a technique created for the Design Space Exploration (DSE) of many-core systems, is introduced in one of the types of research. The authors note that the search process's beginning position has a significant impact on how well simulated annealing performs. In order to address this, they create an ML model that determines the important design space regions to concentrate on, hence minimizing the number of pointless excursions.

Three meta-heuristic algorithms—simulated annealing (SA), genetic algorithms (GA), and ant colony optimization (ACO)—are offered to help determine hyper-parameter settings in a more recent study by Wang and Schäfer [34] using a number of machine learning (ML) techniques. The authors create an ML model for each algorithm that forecasts runtime and design quality (as determined by the hyper-parameter settings and quantified by Average Distance to the Reference Set, or ADRS). Their models show an improvement in ADRS of over 1.92 times when compared to default hyper-parameters with comparable runtimes. Further improving

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7632
***Article Received:** 04 June 2023 **Revised:** 26 July 2023 **Accepted:** 14 August 2023*

search efficiency, the authors integrate SA, GA, and ACO to create a revolutionary design space explorer.

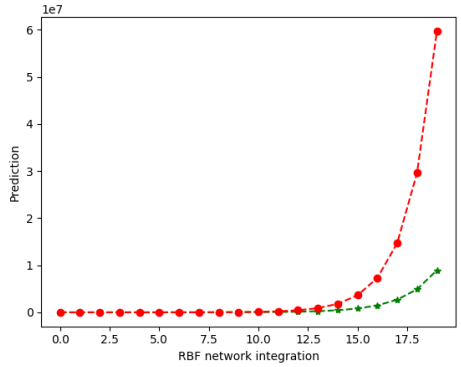

Fig.4 Prediction obtain using RBF network integration

Fig. 4 shows the obtained while using the RBF neural network and from the figure it is obtained that the accuracy of prediction is very high while using the RBFNN. The suggested method consists of two stages: a local search in the first stage that is directed by a goal-based cost function, and a meta search in the second stage that uses the search trajectories from prior local searches to forecast results given particular beginning conditions. Fast Simulated Annealing (FSA) [35], a technique introduced in another paper, uses a decision tree to improve the performance of simulated annealing. In this situation, a common technique for inductive inference is decision tree learning. To create a sufficient training set for building the decision tree, FSA first does conventional simulated annealing. The decision tree is then used to produce new design configurations while keeping the most popular ones.

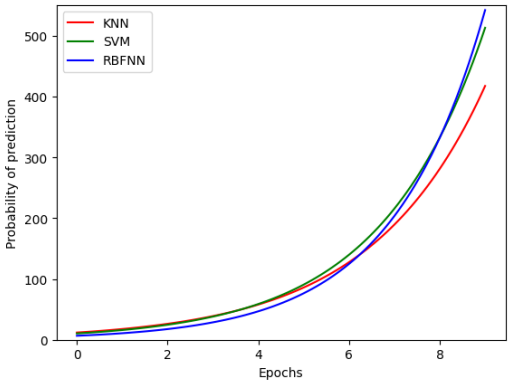

Fig.5 Prediction obtained at each level of Epochs

Fig.5 depicts the prediction obtained at each level of epochs. At each epoch the model will learning from the data. The model is able to learn by itself and the prediction accuracy has been increased drastically. Gram-Schmidt orthogonalization has been used to create a non-iterative and effective weight learning technique that does not require inversion operations. Using this method, the radial basis functions (RBFs) are converted into a collection of orthonormal RBFs, and the best weights are then determined for each orthonormal RBF. Then, without changing the kernel functions, these weights are modified to fit back into the original RBF network topology. As a result, it is no longer necessary to compute the off-diagonal terms in the linear set of weight equations. Additionally, because the weights may be calculated recursively and the computation can be set up in parallel, this method only needs a little amount of storage space. The existing network weights need not be pre computed in order to accommodate new hidden nodes. This makes it possible to train the network effectively by gradually adding hidden nodes until an acceptable error threshold is reached. Additionally, each RBF's unique contribution to the total network output can be evaluated.

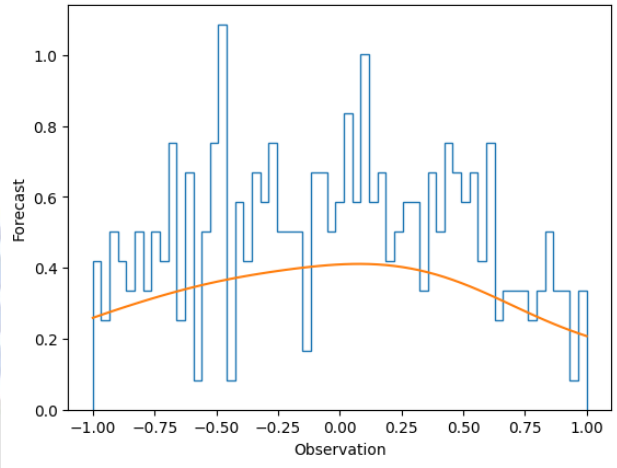

Fig.6 Classification between Observation and Forecast

Fig. 6 depicts the forecast made based on the available data points. 800 data points total from the dataset were split into training and testing datasets. The first 600 samples were used to train the RBF network, and the following 100 samples were used to identify the ideal weighted factor. The last 100 samples were saved for comparing the RBF and PRBF networks' predictive abilities. The minimum embedding dimension or the number of input nodes was determined to be seven using the aforementioned method. The number of centers, which defines the computing needs for each data point, has a direct impact on the RBF network's efficiency. Performance might be improved by using both labelled and unlabeled data as the basis, however this becomes difficult for large-scale datasets. In order to retain optimal performance, a smaller number of centers must be chosen. K-means centers have historically been employed frequently for RBF networks and have proven to function satisfactorily in real-world applications. Recent studies have demonstrated the effectiveness of non-linear features learned by k-means for solving a variety of issues, including optical character

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7632
***Article Received:** 04 June 2023 **Revised:** 26 July 2023 **Accepted:** 14 August 2023*

recognition and visual object detection. We will discuss the benefits of k-means for RBF network centers in this section, as well as how k-means quantization affects the asymptotic characteristics of the RBF method.

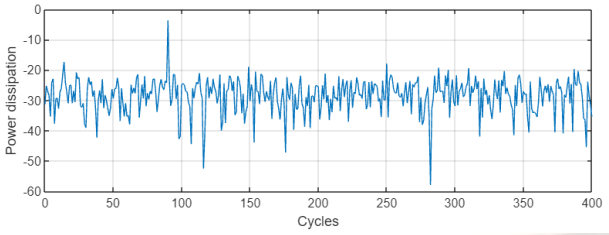

Fig 7. Power dissipation vs Cycles

Fig.7 shows the power dissipation based on the cycles ran. Here also the power dissipation has been considerably minimized. High-level synthesis (HLS) tools can be made more productive by combining machine learning (ML)-based predictors for timing, resource, and latency as well

as by using data-driven searching techniques. This makes it possible to explore a large design space more effectively, which results in the creation of designs of a better caliber. The following procedures are recommended in order to encourage future research in ML used in HLS:

1. Design space exploration (DSE) issues' open benchmark: Due to the high synthesis costs involved in examining a wide design space for each application, it is frequently difficult to evaluate DSE algorithms on more than a few applications. To solve this, developing a benchmark that includes many applications' various implementations will enable a fair assessment of DSE algorithms.

2. Customized ML models: The majority of earlier investigations used pre-made ML models. However, integrating domain expertise and fusing it with common ML methods may improve model performance. For instance, Ustun et al.'s modification of a common graph neural network (GNN) model to address particular delay prediction issues led to increased model accuracy as well as other advantages.

With the goal of facilitating more effective application of ML-based predictors and more effective exploration of the design space, these practices seek to promote improvements in ML research within the context of HLS.

VII. Conclusions

RBF networks and the k-means clustering method surpass MLP networks and the backpropagation method in terms of classification accuracy. The convergence to optimal values is ensured by the linearity of RBF network models, which leads to minimized mistakes with fewer trials, time, and effort. There are numerous methods for estimating power using different tools and technologies, but there aren't any that can be used to estimate VLSI circuit power at the specification level without first knowing how the circuit works. ML models offer a more affordable method of estimating power than EDA tools. Our study makes a contribution by calculating power for both the pre- and post-synthesis phases while taking into account the physical dimensions and connectivity specifics of the transistors.

In order to address difficulties involving nonlinear function learning, we have examined the convergence and convergence rates of normalized radial basis function networks with a single hidden layer. There have also been suggestions made on the choice of radial kernels and their bandwidths. We intend to expand the findings to deep RBF networks in future study and use computer simulations to verify the theory of RBF networks provided in this paper.

The paper should include the following sections: Introduction, Literature Review, Research Methodology, Results, Discussion, Conclusions, Acknowledgments (optional), and References.

References

[1] S. Kim et al., "Middle-of-the-Line Reliability Characterization of Recessed-Diffusion-Contact Adopted sub-5nm Logic Technology", 2022 IEEE International Reliability Physics Symposium (IRPS), pp. 7, 2022

[2] R. Islam, "Early Stage DRC Prediction Using Ensemble Machine Learning Algorithms," in IEEE Canadian Journal of Electrical and Computer Engineering, vol. 45, no. 4, pp. 354-364, Fall 2022.

[3] A. Onan, "Sentiment analysis on massive open online course evaluations: A text mining and deep learning approach", Comput. Appl. Eng. Educ., vol. 29, no. 3, pp. 572-589, May 2021.

[4] M. A. Toçoğlu and A. Onan, "Sentiment analysis on students' evaluation of higher educational institutions", Proc. Int. Conf. Intell. Fuzzy Syst., pp. 1693-1700, 2020

[5] A. Onan, S. Korukoğlu and H. Bulut, "A hybrid ensemble pruning approach based on consensus clustering and multi-objective evolutionary algorithm for sentiment classification", Inf. Process. Manag., vol. 53, no. 4, pp. 814-833, Jul. 2017.

[6] A. Onan and M. A. Toçoglu, "A term weighted neural language model and stacked bidirectional LSTM based framework for sarcasm identification", IEEE Access, vol. 9, pp. 7701-7722, 2021.

[7] S. Pentapati, B. W. Ku and S. Lim, "Machine Learning Integrated Pseudo-3-D Flow for Monolithic 3-D ICs," in IEEE Journal on Exploratory Solid-State Computational Devices and Circuits, vol. 7, no. 1, pp. 35-42, June 2021.

[8] B. W. Ku, K. Chang and S. K. Lim, "Compact-2D: A physical design methodology to build two-tier gate-level 3-D ICs", IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 39, no. 6, pp. 1151-1164, Jun. 2020.

[9] Naik KA, Raj RDA, Rao CV, Babu TS. Generalized cryptographic image processing approaches using integer-series transformation for solar power optimization under partial shading. Energ Conver Manage. 2022

[10] S. S. K. Pentapati, B. W. Ku and S. K. Lim, "ML-based wire RC prediction in monolithic 3D ICs with an application to full-chip optimization", Proc. Int. Symp. Phys. Design (ISPD), pp. 75-82, Mar. 2021.

[11] S. S. K. Pentapati, K. Chang, V. Gerousis, R. Sengupta and S. K. Lim, "Pin-3D: A physical synthesis and post-layout optimization flow for heterogeneous monolithic 3D ICs", Proc. IEEE/ACM Int. Conf.Comput. Aided Design (ICCAD), pp. 1-9, Nov. 2020.

[12] Y. -C. Lu, J. Lee, A. Agnesina, K. Samadi and S. K. Lim, "A Clock Tree Prediction and Optimization Framework Using Generative Adversarial Learning," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 41, no. 9, pp. 3104-3117, Sept. 2022, doi: 10.1109/TCAD.2021.3122109.

[13] S. Koh, Y. Kwon and Y. Shin, "Pre-layout clock tree estimation and optimization using artificial neural network", Proc. ACM/IEEE Int. Symp. Low Power Electron. Design, pp. 193-198, 2020

[14] A. B. Chowdhury, B. Tan, S. Garg and R. Karri, "Robust Deep Learning for IC Test Problems," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 41, no. 1, pp. 183-195, Jan. 2022.

[15] A. F. Tabrizi, N. K. Darav, L. Rakai, I. Bustany, A. Kennings and L. Behjat, "Eh?predictor: A deep learning framework to identify detailed routing short violations from a placed netlist", IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 39, no. 6, pp. 1177-1190, Jun. 2020.

[16] K. Liu et al., "Adversarial perturbation attacks on ML-based CAD: A case study on CNN-based lithographic hotspot detection", ACM Trans. Des. Autom. Electron. Syst., vol. 25, no. 5, pp. 1-31, Aug. 2020.

[17] S. Majzoub, R. Saleh, M. Taouil, S. Hamdioui and M. Bamakhrama, "Rapid Design-Space Exploration for Low-Power Manycores Under Process Variation Utilizing Machine Learning," in IEEE Access, vol. 10, pp. 70187-70203, 2022, doi: 10.1109/ACCESS.2022.3187140.

[18] M. F. Reza, "Machine learning for design and optimization challenges in multi/many-core network-on-chip", Proc. 14th Int. Workshop Netw. Chip Architectures, pp. 29-34, Oct. 2021.

[19] N. Wu, L. Deng, G. Li and Y. Xie, "Core placement optimization for multi-chip many-core neural network systems with reinforcement learning", ACM Trans. Des. Autom. Electron. Syst., vol. 26, no. 2, pp. 1-27, Feb. 2021.

[20] Liu, Z.-H.; Nie, J.; Wei, H.-L.; Chen, L.; Li, X.-H.; Lv, M.-Y. Switched PI Control Based MRAS for Sensorless Control of PMSM Drives Using Fuzzy-Logic-Controller. IEEE Open J. Power Electron. 2022

[21] A. Agnesina, K. Chang and S. K. Lim, "Parameter Optimization of VLSI Placement Through Deep Reinforcement Learning," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 42, no. 4, pp. 1295-1308, April 2023, doi: 10.1109/TCAD.2022.3193647.

[22] Y. -C. Lu, S. Pentapati, L. Zhu, G. Murali, K. Samadi and S. K. Lim, "A Machine Learning-Powered Tier Partitioning Methodology for Monolithic 3-D ICs," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 41, no. 11, pp. 4575-4586, Nov. 2022.

[23] Steven Martin, Thomas Wood, María Fernández, Maria Hernandez, .María García. Machine Learning for Educational Robotics and Programming. Kuwait Journal of Machine Learning, 2(2). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/179

[24] T.Liu,etal.,"Multi-output selective ensemble identification of non-linear and non stationary industrial processes,"IEEE Trans. Neural Networks and Learning Systems,vol.33,no.5,pp.1867–1880,May2022

[25] T. -Y. Hsieh, C. -C. Cheng, W. -J. Chao and P. -X. Wu, "On Development of Reliable Machine Learning Systems Based on Machine Error Tolerance of Input Images," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 42, no. 4, pp. 1323-1335, April 2023.

[26] A. Bochkovskiy, C.-Y. Wang and H.-Y. Liao, "Yolov4: Optimal speed and accuracy of object detection", arXiv:2004.10934, 2020.

[27] Fallon, Elias. "Machine learning in EDA: Opportunities and challenges." 2020 ACM/IEEE 2nd Workshop on Machine Learning for CAD (MLCAD). IEEE, 2020.

[28] Pandey, Manish. "Machine learning and systems for building the next generation of EDA tools." 2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC). IEEE, 2018.

[29] Xie, Zhiyao, Tao Zhang, and Yifeng Peng. "Security and Reliability Challenges in Machine Learning for EDA: Latest Advances." 2023 24th International Symposium on Quality Electronic Design (ISQED). IEEE, 2023.

[30] Govarthan, Praveen Kumar, et al. "Comparative Analysis of Electrodermal Activity Decomposition Methods in Emotion Detection Using Machine Learning." Caring is Sharing–Exploiting the Value in Data for Health and Innovation. IOS Press, 2023.

[31] Koblah, David, et al. "A survey and perspective on artificial intelligence for security-aware electronic design automation." ACM Transactions on Design Automation of Electronic Systems, 2023.

[32] Hassan, Rakibul, et al. "Automated Supervised Topic Modeling Framework for Hardware Weaknesses." 2023 24th International Symposium on Quality Electronic Design (ISQED). IEEE, 2023.

[33] Wang, Peiyong, et al. "Automated quantum circuit design with nested monte carlo tree search." IEEE Transactions on Quantum Engineering , 2023.

[34] Valencia-Ponce, Martin Alejandro, et al. "Integrated Circuit Design of Fractional-Order Chaotic Systems Optimized by Metaheuristics." Electronics, 2023.

[35] Li, Chengxi, et al. "Deep reinforcement learning in smart manufacturing: A review and prospects." CIRP Journal of Manufacturing Science and Technology, 2023.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10s
DOI: https://doi.org/10.17762/ijritcc.v11i10s.7632
***Article Received:** 04 June 2023 **Revised:** 26 July 2023 **Accepted:** 14 August 2023*

[36] Javeed, Ashir, et al. "Machine Learning for Dementia Prediction: A Systematic Review and Future Research Directions." Journal of medical systems, 2023.

[37] Guo, Wenxuan, et al. "Machine learning methods in solving the boolean satisfiability problem." Machine Intelligence Research, 2023.

[38] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan and Q. V. Le, "AutoAugment: Learning augmentation strategies from data", 2021

[39] Sandeep Kadam, & T. Srinivasarao. (2023). ElitGA : Elitism Based Genetic Algorithm for Evaluation of Mutation Testing on Heterogeneous Dataset. International Journal of Intelligent Systems and Applications in Engineering, 11(4s), 509–516. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/2720

[40] Sánchez D, Servadei L, Kiprit G, Wille R and Ecker W. ,A Comprehensive Survey on Electronic Design Automation and Graph Neural Networks: Theory and Applications. ACM Transactions on Design Automation of Electronic Systems, Mar-2023

[41] Li B, Zhang X, You H, Qi Z and Zhang Y. (2022). Machine Learning Based Framework for Fast Resource Estimation of RTL Designs Targeting FPGAs. ACM Transactions on Design Automation of Electronic Systems. 28:2. (1-16), Mar-2023.

[42] Poovannan, E., and S. Karthik. "Power Prediction of VLSI Circuits Using Machine Learning." Computers, Materials & Continua, 2023.

[43] Oh M, Lee K, Kim S and Park B. (2023). Data-driven multi-objective optimization with neural network-based sensitivity analysis for semiconductor devices. Engineering Applications of Artificial Intelligence, Jan-2023

[44] Sahruday, Gaje, Anumula Srikanth, and Karne Harikrishna. "The Power Optimization and an Area Efficient of Static RAM 1-Bit Cell using CMOS Novel Technologies." Journal of Electronics and Informatics, 2023.

[45] Khan, Imran Ullah, Nupur Mittal, and Mohd Amir Ansari. "Applications of VLSI Design in Artificial Intelligence and Machine Learning." Machine Learning for VLSI Chip Design, 2023.

[46] Kumar, Pratik, et al. "Process, bias, and temperature scalable cmos analog computing circuits for machine learning." IEEE Transactions on Circuits and Systems I: Regular Papers, 2022.

[47] Jacinto, Gabriel Lima, et al. "Exploring Machine Learning for Electrical Behavior Prediction: The CMOS Inverter Case Study." 2022 35th SBC/SBMicro/IEEE/ACM Symposium on Integrated Circuits and Systems Design (SBCCI). IEEE, 2022.

[48] Bhavesh, Modi Divy, et al. "Power consumption prediction of digital circuits using machine learning." 2022 2nd International Conference on Artificial Intelligence and Signal Processing (AISP). IEEE, 2022.

[49] Poovannan, E., and S. Karthik. "Pre Synthesis and Post Synthesis Power Estimation of VLSI Circuits Using Machine Learning Approach." Applied Artificial Intelligence, 2022.

[50] Sahruday, Gaje, Anumula Srikanth, and Karne Harikrishna. "The Power Optimization and an Area Efficient of Static RAM 1-Bit Cell using CMOS Novel Technologies." Journal of Electronics and Informatics, 2023.

[51] Khan, Imran Ullah, Nupur Mittal, and Mohd Amir Ansari. "Applications of VLSI Design in Artificial Intelligence and Machine Learning." Machine Learning for VLSI Chip Design, 2023.

[52] Panagiotopoulou, C., et al. "Power Analysis Methodology for Energy Harvesting System on Chip." 2022 Panhellenic Conference on Electronics & Telecommunications (PACET). IEEE, 2022.

[53] Syamala, Y. "Machine learning-based design automation of CMOS analog circuits using SCA-mGWO algorithm." [ETRI] ETRI Journal, 2022.

[54] Olivera, Fabián, and Antonio Petraglia. "Ultra-low-power CMOS voltage references: Analysis and optimization regarding technology node." AEU-International Journal of Electronics and Communications, 2023.