IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 15 May 2023, accepted 5 June 2023, date of publication 9 June 2023, date of current version 16 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3285113

RESEARCH ARTICLE

AMS Circuit Design Optimization Technique Based on ANN Regression Model With VAE Structure

JIN-WON HYUN AND JAE-WON NAM, (Member, IEEE)
Department of Electronic Engineering, Seoul National University of Science and Technology, Seoul 01811, South Korea

Corresponding author: Jae-Won Nam (jaewon.nam@seoultech.ac.kr)

This work was supported in part by the Ministry of Science and Information and Communications Technology (MSIT), South Korea, through the Information Technology Research Center (ITRC) Support Program, Supervised by the Institute for Information and Communications Technology Planning and Evaluation (IITP), under Grant IITP-2023-RS-2022-00156295.

ABSTRACT The advanced design of an analog mixed-signal circuit is not simple enough to meet the requirements of the performance matrix as well as robust operations under process-voltage-temperature (PVT) changes. Even commercial products demand stringent specifications while maintaining the system's performance. The main objectives of this study are to increase the efficiency of the design optimization process by configuring the design process in multiple regression modeling stages, to characterize our target circuit into a regression model including PVT variations, and to enable a search for co-optimum design points while simultaneously checking performance sensitivity. We used an artificial neural network (ANN) to develop a regression model and divided the ANN modeling process into coarse and fine simulation steps. In addition, we applied a variational autoencoder (VAE) structure to the ANN model to reduce the training error due to an insufficient input sample. According to the proposed algorithm, the AMS circuit designer can quickly search for the co-optimum point, which results in the best performance, while the least sensitive operation as the design process uses a regression model instead of launching heavy SPICE simulations. In this study, a voltage-controlled oscillator (VCO) is selected to prove the proposed algorithm. Under various design conditions (CMOS 180 nm, 65 nm, and 45 nm processes), we proceed with the proposed design flow to obtain the best performance score that can be evaluated by a figure-of-merit (FoM). As a result, the proposed regression model-based design flow achieves twice accurate results in comparison to that of the conventional single-step design flow.

INDEX TERMS Analog circuit design automation, variational-autoencoder, regression model, artificial-neural-network, voltage-controlled-oscillator.

I. INTRODUCTION

With advancements in semiconductor processing technology, high-performance digital processing can be applied for reasons such as significant power loss reduction, high operating frequency, and area reduction. The finer the process, the lower the supply voltage that can be applied, and the shorter the minimum length of the transistor. And power consumption is reduced by the square of the supply voltage, and the area is reduced by shortening the transistor length. Also, in the case of digital circuits, the performance of transistors improves as the processing unit decreases. Therefore, as the process technology has developed to a few nanometers, the performance of digital circuits has improved significantly. In addition, even if the process is different, the logic structure of the digital circuit has the same structure, and because the digital circuit has binary inputs and outputs, it is robust against noise. For this reason, if only a standard cell is made, various circuits can be easily implemented because there are few variables to be adjusted.

The associate editor coordinating the review of this manuscript and approving it for publication was Ludovico Minati.

Digital circuit design has standardized design automation with a programming language. Because the circuit is generated by matching it with the programming language, even if the processing unit is different, it can be expressed at the same logic level. This has the advantage of easy migration of the circuits to another process. Design automation is made possible by the synthesis step that converts the HDL code into a digital logic circuit, and the place and route steps that design the layout using algorithms such as the Kernighan-Lin algorithm or Fiduccia-Mattheyses algorithm. Based on this design flow, many design automation algorithms have been implemented for digital circuit design [1].

The characteristics and performance of Metal-Oxide-Silicon Field-Effect-Transistor (MOSFET) vary depending on the process technology node. In addition, the mismatch between transistors becomes more severe as the process progresses to a smaller length of the MOSFET [2]. Accordingly, when designing an analog circuit, correction circuits that compensate for this mismatch are added. In the case of the latest process, the correction circuit is almost indispensable owing to a severe mismatch. Because of various other problems, analog design flow is redesigned based on the designer's experience [1]. This is because in the heuristic design method, even experienced designers may not recognize or misplace optimization points. In addition, analog circuit design has many variables to consider compared to digital circuit design, so there are many difficulties in developing the automated design of analog circuits. Although research on the automation of analog circuit design is in progress, much more computer processing time is required than digital design time.

The purpose of this research is to reduce the time required to automate the optimization design with Analog and Mixed-Signal (AMS) circuits and to obtain more accurate results. We implemented an algorithm for automatic circuit design utilizing the zoom-in algorithm proposed by Hyun and Nam [3], along with a regression model and a variational autoencoder structure. The proposed algorithm is applied to the automated design of a voltage-controlled oscillator (VCO) that consists of a ring oscillator circuit. It is designed to verify the range of design parameters that the designer may not recognize. If the amount of data is increased based on the proposed algorithm, high accuracy can be obtained. However, we aim to design automation with less computer processing time than that in previous studies. To realize this, we train a regression model using the zoom-in algorithm. And we implement to reduce the error of the regression model through a variational autoencoder (VAE) structure. Moreover, a verification step is added to robustly design process-voltage-temperature (PVT) variations.

In this paper, Section II describes the structure of the regression model and key optimization algorithms. In Section III, the design optimization flow is explained in detail, and in Section IV, the simulation setup is illustrated for this work. In Section V and Section VI, we present the simulation results and conclusions, respectively.

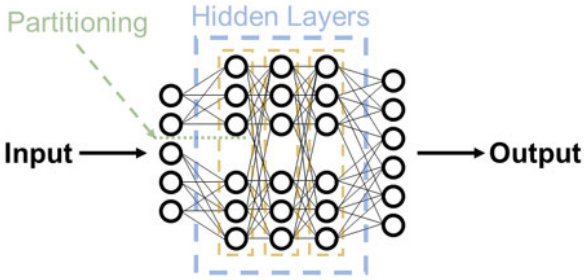


FIGURE 1. ANN regression model structure.

II. METHOD

In analog circuit design automation, reducing computer processing time is an important task. if it is much slower than the designer's direct optimization design time without design automation, there is no advantage to design automation. Therefore, to reduce the computer processing time, the machine learning model structure and algorithms are developed.

A. ANN REGRESSION MODEL STRUCTURE

In this study, a regression model trained using simulation data finds an optimization point. The simulation data consist of the results concerning specific points discretely partitioned within the entire range. A parametric search through SPICE simulation shows detailed performance results over the entire range. However, the design time will increase infinitely. To address the issue of time constraints, we utilize a regression model that interpolates simulation results at desired design points. Furthermore, considering the limited availability of training data for the regression model, an Artificial Neural Network (ANN) model is deemed more appropriate than a deep learning model like a Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), or Deep Neural Network (DNN) [4]. The process of training the ANN regression model is optimized using the MSE-type loss function and the Adam optimizer, as defined in Algorithm 1.

Fig. 1 illustrates the structure of the ANN regression model. It consists of three layers: an input layer, a hidden layer, and an output layer. The input layer represents the design parameters, while the output layer represents the desired design specifications. The hidden layer refers to the layer excluding the input and output layers. The size of the hidden layer depends on the dataset size. If the dataset size significantly differs from the ANN model size, it can lead to accuracy issues like overfitting or underfitting. To address this, it is advisable to adjust the size of the ANN model proportionally to the dataset size, ensuring better accuracy. Additionally, due to varying weights for each input-output relationship, we partitioned the hidden layer connected to the input and output of the ANN regression model. This partitioning helps avoid biased learning towards a specific design parameter and provides separate training guidelines for input and output components.

Algorithm 1 Training ANN Regression Model

Input: $P = \{p_1, p_2, \ldots, p_N\}$
▷ Dataset containing N design parameters
Output: $R = \{r_1, r_2, \ldots, r_M\}$
▷ Dataset containing M performances

Initialize weights and data standardization
for epoch in $1, 2, \ldots, e$ **do**
  compute the loss function $L = \frac{1}{2}\Sigma_{i=1}^{b}(y_i - t_i)^2$
  for mini-batch in $1, 2, \ldots$ **do**
    optimize weights by Adam optimizer
  end for
  b is a mini-batch size
  y_i is a prediction of output
  t_i is a validation of output
end for

B. ZOOM-IN ALGORITHM

When designing a circuit, designers optimize it through numerous trials and errors. Additionally, without utilizing a machine learning-based learning model, a significant amount of data covering the global range is required to find the optimal point. To enhance the design optimization process, our proposed zoom-in algorithm in this study achieves greater precision and efficiency compared to previous research efforts. While previous studies have explored the use of regression models for analog circuit design automation [3], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], the original zoom-in algorithm presented in [3] had a limitation in the accuracy of the regression model during the initial step due to a smaller amount of data compared to single-step approaches. To address this limitation, our study focuses on improving accuracy by incorporating the variational autoencoder (VAE) structure and refining the verification step within the algorithm.

The proposed zoom-in algorithm is divided into a coarse simulation in a wide sweep range and a fine simulation in a narrow sweep range. A coarse simulation step does not require high accuracy because it only needs to determine the rough optimal points. The coarse simulation step properly lists up to candidate groups based on the trend of change in the result values. In contrast, the purpose of the fine simulation is to perform precise optimization by equally performing each predicted point within a small range. If the model is trained with a large dataset without dividing the steps, the amount of data required increases exponentially, and the processing time can be infinitely long. Consequently, if the optimization steps are divided, the time required to compute unnecessary simulation results can be saved. Although the zoom-in algorithm can be divided into multiple stages instead of 2 stages, assuming the same amount of data, there is a high possibility that it can be optimized to the local optima since the accuracy of the first stage is lower than previously. Alternatively, it is suitable for cases in which there are many parameters to be considered or optimized over a wider range.

In this study, the zoom-in algorithm sets the number of simulations of the narrow sweep range by $1/(N+1)$ times the total number of simulations for a wide sweep range. N is the number of narrow sweep range (Narrow S.R) candidates. That is, the size of the training dataset is set to be the same for both fine and coarse simulation steps. Fig. 2 illustrates the range settings for the coarse and fine simulation steps in the zoom-in algorithm. After selecting the best candidate group in the coarse simulation, the range is divided during the fine simulation. With 2 design parameters and 5 candidates for the fine simulation, both the fine and coarse simulations involve a dataset size of (6×6), resulting in 216 data points for optimizing the design parameters. In contrast, the single-step approach would require a larger dataset size of $(6 \times 6) \times (6 \times 6)$ when assuming a 10 % fine simulation range. This would amount to 1296 data points needed for optimization. The zoom-in algorithm significantly reduces the dataset size, leading to improved efficiency and computational savings.

The ranges of the fine simulations vary for each candidate, as they are specified based on the error rate compared to the results of the SPICE simulation and the regression model. More specifically, the range of the fine simulation is selected to be between 5 % and 15 % of the range of the coarse simulation. This approach compensates for poorly trained regions in the regression model, ensuring they are not discarded.

C. REFLECTING PVT VARIATION

The analog circuit designer selects the optimization point with the best Figure-of-Merit (FoM) performance when designing the circuit. However, as the process advances, the mismatch between devices increases. Due to this, designing an analog circuit is more difficult than before. Although prediction techniques such as Monte Carlo have been created, the Monte Carlo simulation takes significant time for processing a detailed design [16]. To replace this, we predicted the value by applying limited PVT variations through a regression model and verification step.

Analog designers can be very good at optimizing designs; however, they tend to rely on their experience. Even skilled designers can select different design points for the same circuit. In contrast, the regression model finds better points that the designer does not consider. To be more specific, the regression model searches points with similar performance metrics. In each of the searched points, while the performance metrics are similar, the variations in performance resulting from mismatches differ. Therefore, it is necessary to optimize the design values for various points and compare the PVT variations. Fig. 3 shows an example of how the FoM can vary by changing the design parameters. The result of point B is larger than that of point A. In contrast, the same performance as the reference point cannot always be obtained, since there are mismatches between processes. Both points A and B are given the same change in parameter values, but the amount of change in the FoM is different. If the mismatch between processes is considered, the designer will choose point A.

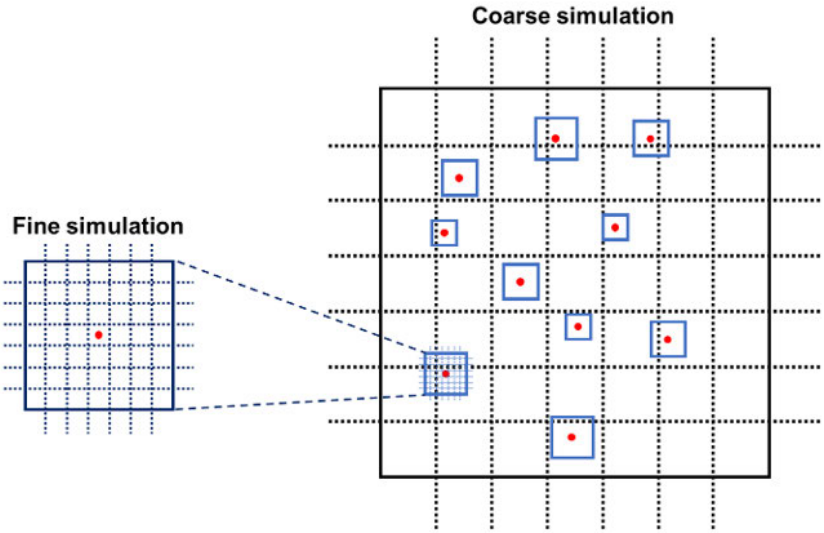


FIGURE 2. Example of zoom-in algorithm about 2-axis design parameter.

Therefore, considering the PVT variations, optimization with simulation results over a wide range can design a circuit with better performance. Similarly, studies including corner simulation, layout generator, and Monte Carlo simulation have been developed to consider PVT variations [10], [11], [12], [13], [14]. These simulations to check PVT variations have the disadvantage of requiring a significant computer processing time. In this study, the PVT variations are considered with less computer processing time. We directly derived the results for each variance using the trained regression model, and the method for restrictively applying the PVT variations is described in detail in Section III.

D. APPLYING VAE STRUCTURE

We selected candidates from a wide range using a zoom-in algorithm. Although studies that have applied automated design using an ANN-based regression model already exist, the regression model may be trained partially incorrectly. This is because it predicts values based on limited simulation data. Therefore, to solve this problem, we predict values over a wide range using an ANN-based regression model and apply additional compensation methods. We created another model that is trained by reversing the input and output. Fig. 4 shows the VAE structure and the structure of the proposed method. In Fig. 4(a), the VAE is divided into the encoder and decoder parts.

The VAE is a type of generative model like GAN (Generative Adversarial Networks (GAN) and diffusion models [17]. It is an advanced model of an autoencoder (AE) that creates a meaningful latent space by making the input and output equal and derives the latent space using an encoder and a decoder. The desired output is decoded from this latent space for data generation. The VAE aims to approximate the true distribution of input data $p(\mathbf{x})$. To this end, the VAE consists of an encoder, a decoder, and a latent space. The encoder is responsible for transforming the input into latent space. The purpose of the encoder is to estimate the distribution of the latent space vector $\mathbf{z}$, that is, $q(\mathbf{z}|\mathbf{x})$. Find the parameters of the mean (μ) and standard deviation (σ) by selecting the normal distribution that best represents $q(\mathbf{z}|\mathbf{x})$. In contrast to an encoder, a decoder is responsible for transforming a latent space into an input. The goal of the decoder is to estimate $q(\mathbf{z}|\mathbf{x})$ given the latent space vector $\mathbf{z}$ as the input. Because it generates data $\mathbf{x}$ again according to a given vector $\mathbf{z}$, the decoder serves as a generative model. Latent space refers to any hidden vector. Unlike autoencoders, a VAE samples noise and creates a latent space to generate data. The VAE finds θ that maximizes $p_\theta(\mathbf{x})$ using a maximum likelihood estimation (MLE) approach. Equation (1a)–(1d) are the process for maximizing the log-likelihood through the MLE approach.

$$\log p_\theta(\mathbf{x}) = \int q_\phi(\mathbf{z}|\mathbf{x}) \log p_\theta(\mathbf{x}) d\mathbf{z} \tag{1a}$$

$$= \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{p_\theta(\mathbf{z}|\mathbf{x})} d\mathbf{z} \tag{1b}$$

$$= \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{p_\theta(\mathbf{z}|\mathbf{x})} \frac{q_\phi(\mathbf{z}|\mathbf{x})}{q_\phi(\mathbf{z}|\mathbf{x})} d\mathbf{z} \tag{1c}$$

$$= \int q_\phi(\mathbf{z}|\mathbf{x}) \log p_\theta(\mathbf{x}) d\mathbf{z} - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) + \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z}|\mathbf{x})) \tag{1d}$$

$$\geq \int q_\phi(\mathbf{z}|\mathbf{x}) \log p_\theta(\mathbf{x}) d\mathbf{z} - \mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) \tag{1e}$$

In (1b), Bayes' rule is applied in (1a), and (1c) is multiplied by the same q equation for the denominator and numerator in (1b). In (1d), Kullback–Leibler (KL) divergence in (1c) is applied. The first term in (1d) is equal to the negative cross-entropy between p(x|z) and q(z|x). The second term is equal to the KL divergence between q(z|x) and p(z). The

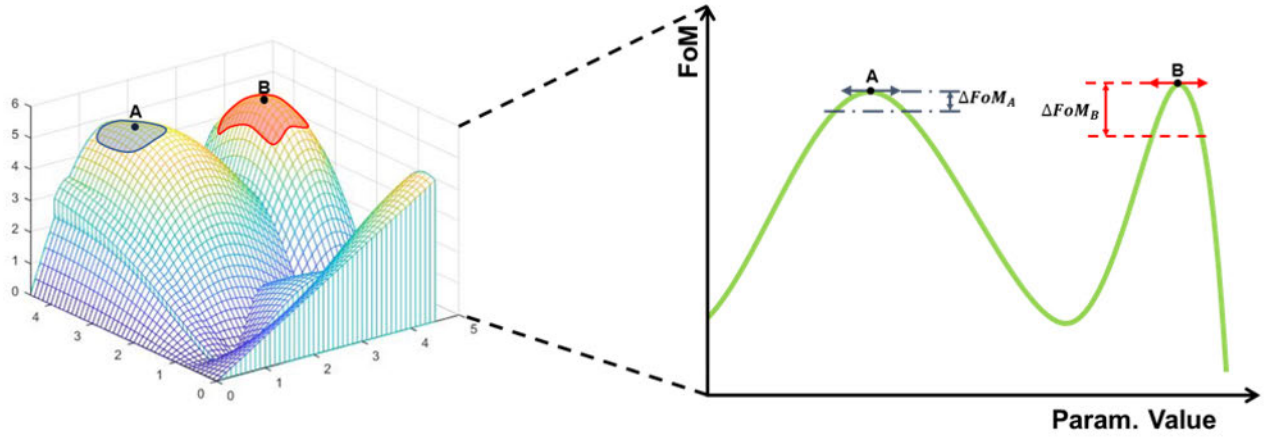


FIGURE 3. Apply PVT variation using results of ANN regression model.

last term cannot be computed because p(z|x) is unknown. However, because KL divergence is always positive, the last expression can be expressed as an inequality, as in (1e). The right-hand side of inequality (1e) is referred to as evidence lower bound (ELBO). By maximizing this ELBO, the log-likelihood can also be maximized. After learning the encoder and decoder while maximizing the aforementioned ELBO, the desired data can be generated using the learned decoder.

Subsequently, to the employment of all the above-mentioned terms, our model finds other points with similar results to the design optimization point. Therefore, based on this concept, we trained 2 models to predict inputs as well as the traditional model to predict outputs, as shown in Fig. 4(b). Using this model, we add validation of the preexisting results. And we find new design parameter candidates that are obtained using the preexisting results. In other words, we also added candidates by adding random noise to find the parameters in other locations with similar results.

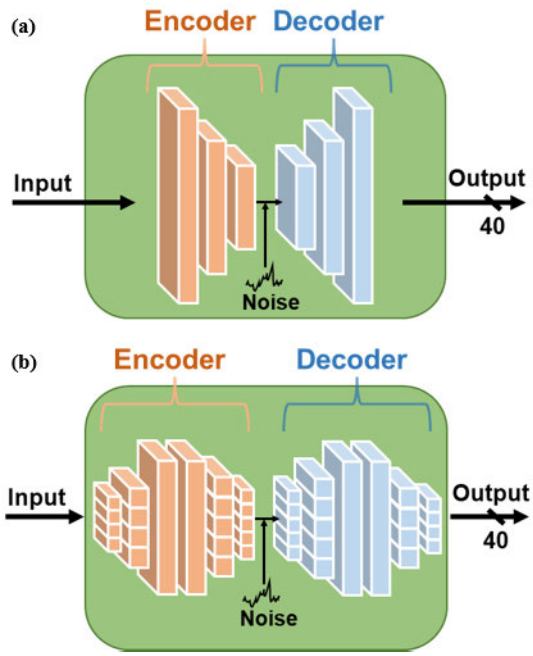


FIGURE 4. (a) General VAE structure (b) The VAE structure employed in this work.

III. DESIGN FLOW

Fig. 5 shows the overall automation design flow used in this study. This section details each step in the design automation flow.

A. COARSE SIMULATION (WIDE RANGE SEARCHING)

First, we need to determine how to divide each design parameter. Each design parameter is divided over a wide range. A representative dividing method randomly or uniformly divides the sections. In this study, each design parameter is evenly divided over a wide sweep range based on the set value. This is because, if the simulation results are small, random division can be a problem. A method that randomly divides the range about small simulations can be biased to one side and split. This is because in that case, the simulation results are not linear, and the accuracy of the regression model for the opposite side can be very small. On the other hand, we just need to obtain a rough location for the largest FoM with small simulations. By employing uniform division, the evenly distributed data across the entire range reduces the occurrence of poorly trained regions. Therefore, a uniform division with small simulations is the most reasonable way to train a regression model.

B. TRAINING VAE STRUCTURE REGRESSION MODEL

As mentioned in Section II-C, we should not simply find the point with the highest performance specification but also find a point that is robust against a process mismatch. We employed the VAE structure to validate the predicted results or to find a new point. The purpose of adding random noise is to find other points that have similar results.

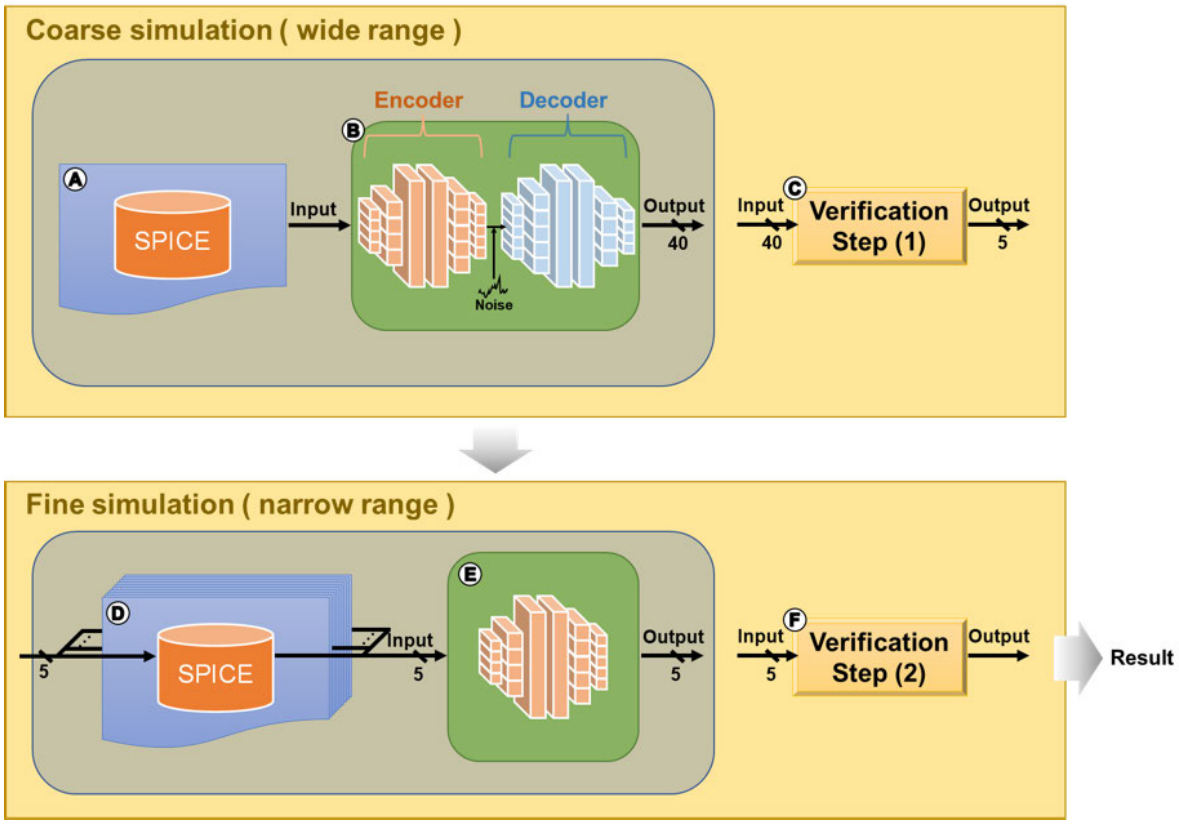


FIGURE 5. Overall automation design flow.

In this study, a regression model of a VAE structure, such as that shown in Fig. 4 is trained using the coarse simulation results. The trained model predicted all outcomes within a wide sweep range. And the 10 best-performing candidates are selected by comparing the predicted results through the encoder. In this process, the results around the candidate point are initialized each time a candidate group is selected such that the sweep range between the candidates does not overlap. Thereafter, we take 10 candidates as input to the decoder part and add only decoder outputs that do not overlap with existing candidates. And random noise is added to each of the 10 candidates and fed into the model. Similarly, only candidates that do not overlap with the existing candidate group are added.

C. VERIFICATION STEP (I)

Using the VAE structure, we obtained a maximum of 40 candidates from the existing 10 candidates. Because 40 candidates is a large number to train into each regression model, the best-performing point must be predicted and reselected through this step. First, we computed only the top 10 candidates with the best FoM performance out of 40 candidates. Then, for the remaining 10 candidates, simulations are conducted under Slow-Slow (SS) and Fast-Fast (FF) transistor conditions. According to the simulations, the final 5 candidates with the smallest performance variation are selected. Finally, regarding the final 5 candidates, the simulation and prediction results are compared, and the narrow sweep range is adjusted in proportion to the error rate in the range of 5 to 15 % of the wide sweep range. A detailed description of this step is presented in Algorithm 2.

D. FINE SIMULATION (NARROW RANGE SEARCHING)

First, we shortened the design time by running 10 SPICE programs simultaneously and proceeding with fine simulations. The same amount of dataset is extracted as a coarse simulation. And the process of extracting the overall simulation results is handled in the same manner as for the coarse simulation. Similarly, when the range is evenly divided at the fine simulation, a much more accurate prediction is possible than at the coarse simulation. As the distance of input parameters between each result is smaller than before.

E. TRAINING REGRESSION MODELS WITH FINE SIMULATION

The regression models are trained to interpolate each narrow sweep range. The structure of the regression models is the same as that of the encoder in the VAE structure. The narrow sweep range is a very small area compared to the wide sweep range, and the number of simulation data points is the same. That is to say, the accuracy of the predicted results derived

Algorithm 2 Verification Step (I) Algorithm (Section III-C)

Input: $I = \{i_1, i_2, \ldots, i_N\}$
▷ N candidates of fine simulation step
Output: $O = \{o_1, o_2, \ldots, o_M\}$
▷ M selected ranges of fine simulation step

Step 1: FoM comparision
$i_i = \langle p_{i,1}, p_{i,2}, \ldots, p_{i,U} \rangle$
▷ $p_{i,j}$: The j^{th} design parameter or performance of i_i
▷ U : The number of design parameters and performances
$H = \{h_1, h_2, \ldots, h_K\}$
▷ h_n : i_i with the n^{th} best FoM
▷ K : The number of selected candidates at step 1
procedure
  for $j \leftarrow 1$ to N **do**
    Compute $p_{i,j}$ from SPICE with TT condition
    Calculate FoM_j
  end for
  $H \leftarrow \text{select}(p_{i,j}, \text{FoM}_j, K)$
  ▷ select: A function that selects and returns the K lowest $p_{i,j}$ based on FoM_j
end procedure

Step 2: Temperature variation comparison
$o_i = \langle q_{i,1}, q_{i,2}, \ldots, q_{i,U} \rangle$
▷ $q_{i,j}$: The j^{th} design parameter or performance of o_i
▷ U : The number of design parameters and performances
$V = \{v_1, v_2, \ldots, v_K\}$
▷ The variation of FoM of h_k
$H = \{h_1, h_2, \ldots, h_K\}$
▷ h_n : i_i with the n^{th} best FoM
procedure
  for $j \leftarrow 1$ to K **do**
    Compute $q_{i,j}$ from SPICE with SS and FF condition
    Calculate v_j
  end for
  $O \leftarrow \text{select}(h_{i,j}, v_j, M)$
  ▷ select: A function that selects and returns the M lowest $h_{i,j}$ based on v_j
end procedure

Step 3: Sweep range setting
procedure
  Calculate error rate
  Set sweep range based on the error rate
end procedure

Algorithm 3 Verification Step (II) Algorithm (Section III-F)

Input: $D = \{d_1, d_2, \ldots, d_T\}$
▷ Dataset containing T results of each ANN regression model
▷ T : The number of predicted results
Output: $Y = \{y_1, y_2, \ldots, y_E\}$
▷ E selected values of design parameters
▷ E : The number of design parameters

Step 1: Select the best FoM of each model
$d_i = \langle w_{i,1}, w_{i,2}, \ldots, w_{i,U} \rangle$
▷ $w_{i,j}$: The j^{th} design parameter or performance of d_i
▷ U : The number of design parameters and performances
$F = \{f_1, f_2, \ldots, f_K\}$
▷ f_n : d_i with the n^{th} best FoM
procedure
  for $i \leftarrow 1$ to M **do**
  ▷ M : The number of fine simulation candidates
    for $j \leftarrow 1$ to T **do**
      Calculate FoM_j
    end for
    $f_i \leftarrow \text{select}(d_{i,j}, \text{FoM}_j, 1)$
    ▷ select: A function that selects and returns the lowest $d_{i,j}$ based on FoM_j
  end for
end procedure

Step 2: PVT variation comparison
minFoM_i : The minimum FoM of i^{th} candidates
maxFoM_i : The maximum FoM of i^{th} candidates
procedure
  for $i \leftarrow 1$ to M **do**
    for $j \leftarrow 1$ to T **do**
      if $(f_i - d_i)/f_i * 100\% \leq 5\%$ **then**
        Calculate FoM_j
        if $\text{FoM}_j \leq \text{minFoM}_i$ **then**
          $\text{minFoM}_i \leftarrow \text{FoM}_j$
        end if
        if $\text{maxFoM}_i \leq \text{FoM}_j$ **then**
          $\text{maxFoM}_i \leftarrow \text{FoM}_j$
        end if
      end if
    end for
    $Y \leftarrow \text{select}(f_i, \text{maxFoM}_i - \text{maxFoM}_i, 1)$
    ▷ select: A function that selects and returns the lowest f_i based on $(\text{maxFoM}_i - \text{maxFoM}_i)$
  end for
end procedure

from the regression models is higher than a regression model that is trained at the coarse simulation step. Moreover, we just use the encoder part of VAE-structured to verify the results. The VAE structure is an additional method to compensate for or verify the accuracy of the model owing to problems; therefore, it is not required an entire part of the VAE structure in the fine simulation step. Consequently, we train each regression model using the encoder part. And the point with the highest FoM is selected based on each regression model.

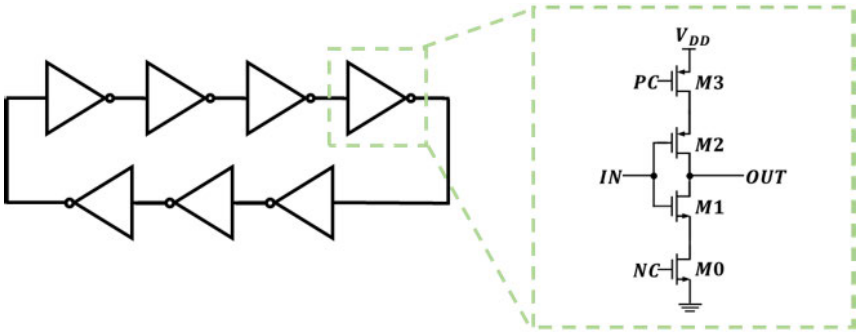


FIGURE 6. Designed VCO block diagram and schematic circuit.

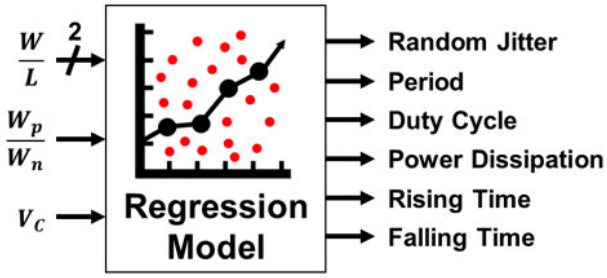


FIGURE 7. ANN regression model's input and output.

F. VERIFICATION STEP (II)

In this step, data processing is performed in a flow similar to that described in Section III-B. Although a mismatch can be applied for each element to predict the precise PVT variations, this method is time-consuming. Furthermore, in the verification step (II), calculations need to be performed for PVT variations of 5 candidates. To minimize processing time, the approach described in Section II-C is employed to select candidates with the least variation. Specifically, the design parameters of the optimized point calculated within each narrow sweep range are altered by 5 %. The point exhibiting the least variation, as determined by the FoM result, is then chosen as the final result. The final result is optimized; however, the designer may need to make slight adjustments for subsequent layout design. Additionally, it is recommended to perform simulations to verify PVT variations before the layout design phase. The detailed flow of the algorithm for this step is described in Algorithm 3.

IV. CIRCUIT DESIGN

In this study, an algorithm for the automated design of a 1 GHz VCO is implemented, and the circuit is optimized. The FoM is calculated using the root-mean-square (RMS) value of the random jitter and power dissipation.

A. VOLTAGE-CONTROLLED-OSCILLATOR (VCO)

The VCO circuit generates an actual clock from the Phase-Locked-Loop (PLL) circuit. The clock frequency changes in proportion to the size of the voltage. We designed a ring oscillator that is connected to a 7-stage inverter. Because the LC oscillator has a standardized design flow, the ring oscillator is designed to be more suitable for automated design. The inverter is a current-starved structure, as shown in Fig. 6. The NC and PC wires are connected to a diode-connected transistor. In other words, the current-starved inverter uses the current mirrors of the PMOS and NMOS to control the charge flow and change the frequency. For the V_{DD}, power supply voltages of 1 V, 1.2 V, and 1.8 V are applied to the 45 nm, 65 nm, and 180 nm processes, respectively.

B. DESIGN PARAMETER

Fig. 7 illustrates the parameters used in the design of a current-starved inverter. The control voltage (V_C) is an important parameter as it determines the frequency of the output signal in the VCO circuit. The ratio of PMOS to NMOS (W_p/W_n) significantly affects the duty cycle of the VCO. The W/L ratio, representing the width and length ratio of the transistor, determines the frequency range generated by the VCO circuit. To account for PVT variations, we introduce process variations by examining the differences in Figure of Merit (FoM) resulting from changes in the PMOS, NMOS, and W/L ratios. We also consider voltage variations by comparing the FoM differences based on control voltage variations. Furthermore, temperature variations are incorporated through corner simulations in the verification step (I).

C. SIMULATION ENVIRONMENT

Fig. 7 shows the input-output relationship of the regression model to be trained. The result derives the random jitter, average cycle, duty cycle, power dissipation, rising time, and falling time. We adopt the FoM_J (Figure of Merit for jitter-power based on the PLL) proposed by Gao et al. [18] and replace the overall power dissipation of the PLL with the power dissipation of the VCO (P_{VCO}). Thus, we utilize the revised VCO FoM as follows:

$$\text{FoM} = 10 \log \left(\left(\frac{\sigma_{rms}}{1s} \right)^2 \left(\frac{P_{VCO}}{1mW} \right) \right), \tag{2}$$

where σ_{rms} is the value for rms jitter. A VCO circuit is the most important operating reference for synchronous circuits.

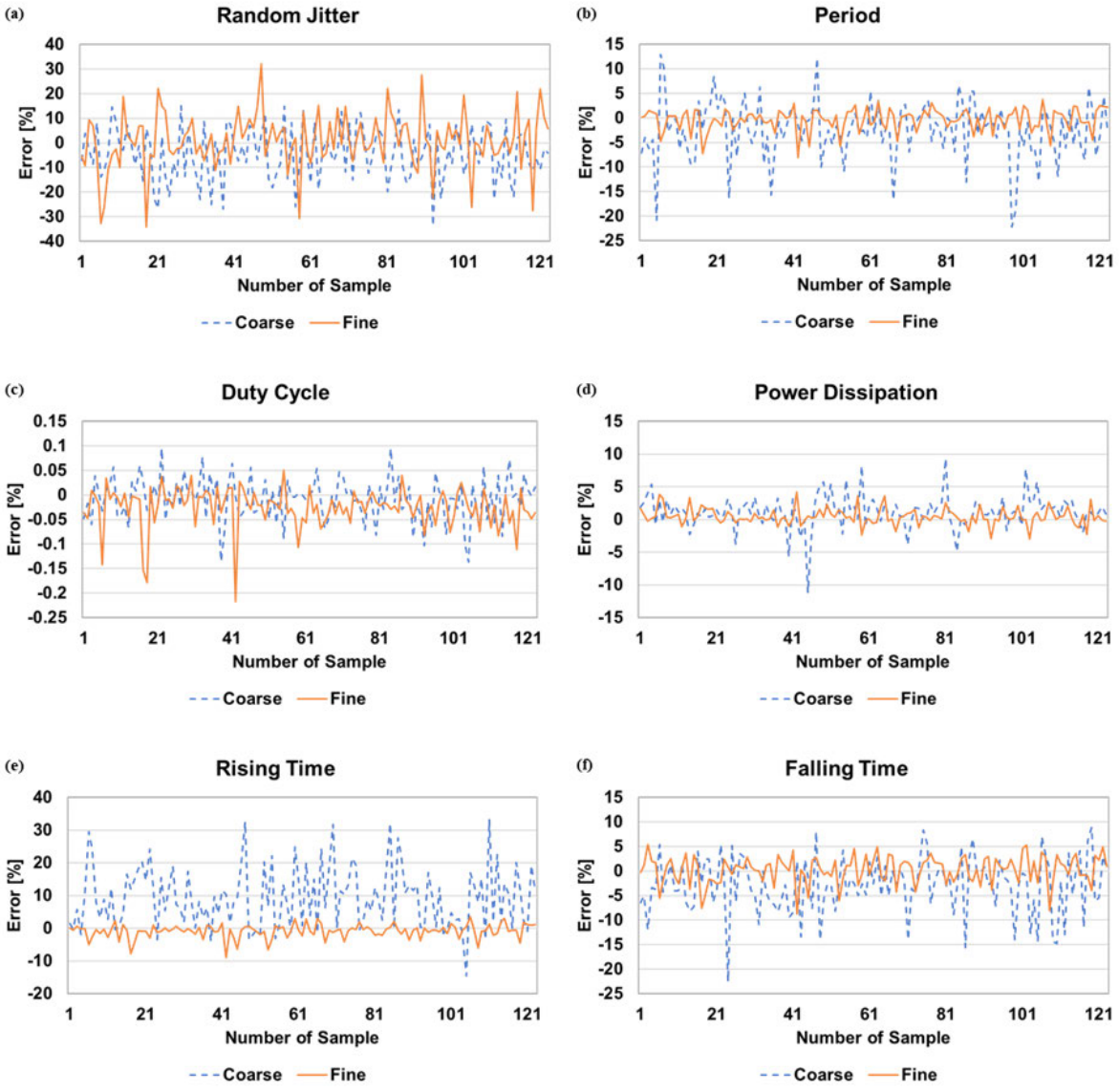


FIGURE 8. ANN regression model accuracy comparison between coarse and fine simulations in CMOS 65nm.

If the clock generated by a clock generator differs significantly from the average period of the clock, the probability of malfunction increases. Power consumption is an important performance specification for every circuit. Therefore, the performance result is compared with the value of the product of the random jitter and power dissipation, as shown in (2). Regarding the remaining parameters, it is not a criterion for selecting the optimized point, but the average cycle, duty cycle, rising time, and falling time are required to verify a circuit that operates normally.

D. SELECTION FILTER

When the design of analog circuits is automated, specifications other than the FoM may not be considered. Taking the VCO circuit as an example, it is not possible to find a proper design point, such as when an operating frequency of 1 GHz cannot be obtained or when the difference in the duty cycle is affected to the circuit. Therefore, a selection filter is required that prioritizes the FoM, but excludes out candidates with specifications below the standard in constructing the rest of the circuit. We use a selection filter to determine whether the circuit is operating normally with the rest of the parameters except the FoM. And select the best result through random jitter and power dissipation.

E. SIMULATION SETUP

The simulation result is derived from the transition time analysis of the SPICE simulation, and it is used to optimize with the goal of a 1 GHz VCO that has low jitter and power dissipation. Additionally, the setup to add noise

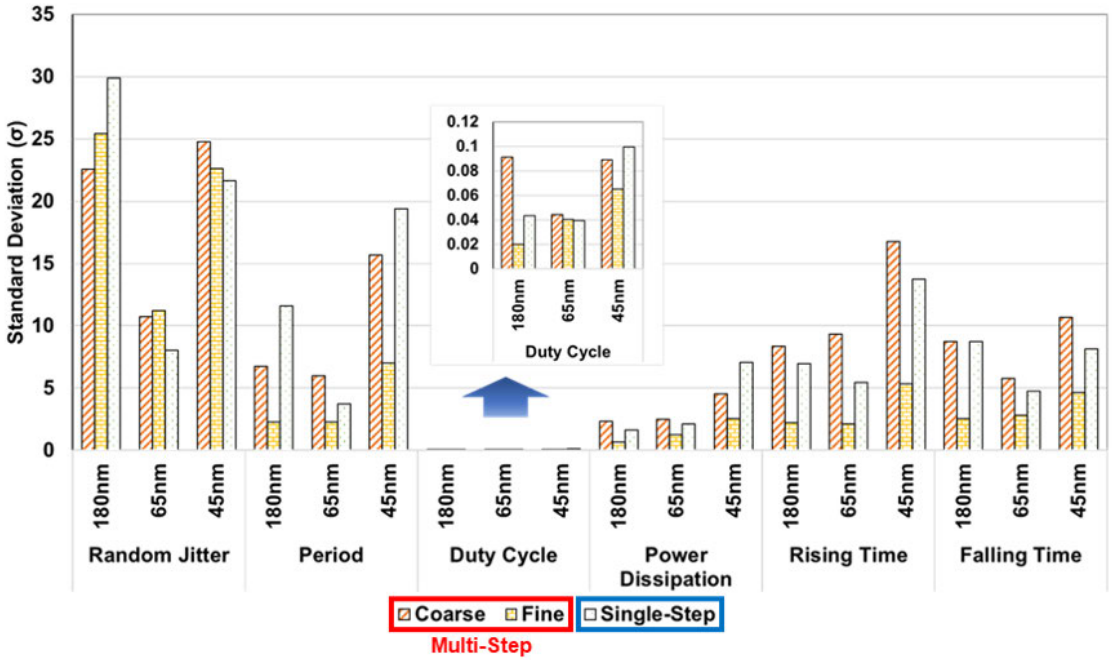


FIGURE 9. ANN regression model standard deviation of error comparison about multi-step (coarse, fine) and single-step simulation.

TABLE 1. Accuracy of the final result in CMOS 180nm, 65nm, and 45nm processes.

CMOS Process	Performance Index	Single-Step			Proposed Multi-Step		
		Regression Model	SPICE	Error	Regression Model	SPICE	Error
180nm	Random Jitter (ps)	0.302	0.494	38.87 %	0.321	0.314	2.23 %
	Period (GHz)	0.998	0.940	6.17 %	1.009	1.031	2.13 %
	Duty Cycle (%)	49.545	55.137	10.14 %	49.507	49.514	0.01 %
	Power Dissipation (mW)	0.346	0.34	1.76 %	1.349	1.343	0.45 %
	Rising Time (ps)	139.09	55.66	149.89 %	209.94	208.32	0.78 %
	Falling Time (ps)	92.90	75.95	22.32 %	101.27	105.93	4.40 %
	FoM (dB)	-255.01	-250.81	4.20 (dB)	-248.57	-248.78	-0.21 (dB)
65nm	Random Jitter (ps)	0.624	0.709	11.99 %	0.581	0.608	4.44 %
	Period (GHz)	0.999	1.020	2.06 %	0.991	0.966	2.59 %
	Duty Cycle (%)	50.375	52.642	4.31 %	50.424	50.417	0.01 %
	Power Dissipation (mW)	0.427	0.424	0.71 %	0.464	0.470	1.28 %
	Rising Time (ps)	158.63	102.99	54.02 %	167.16	164.03	1.91 %
	Falling Time (ps)	154.43	147.55	4.66 %	141.90	140.16	1.24 %
	FoM (dB)	-247.79	-246.71	1.08 (dB)	-248.05	-247.60	0.45 (dB)
45nm	Random Jitter (ps)	2.925	3.274	10.66 %	2.665	2.744	2.88 %
	Period (GHz)	0.996	0.933	6.75 %	0.999	0.991	0.81 %
	Duty Cycle (%)	49.510	49.349	0.33 %	49.662	49.512	0.30 %
	Power Dissipation (mW)	0.0438	0.0445	1.57 %	0.0594	0.0555	7.03 %
	Rising Time (ps)	252.68	270.44	6.57 %	222.67	250.66	11.17 %
	Falling Time (ps)	125.54	128.91	2.61 %	121.53	118.06	2.94 %
	FoM (dB)	-244.26	-243.21	1.05 (dB)	-243.75	-243.79	-0.04 (dB)

is required, because the FoM is calculated with rms jitter. In the case of rms jitter rather than peak-to-peak jitter, there is no significant difference, even if the transition time is long. Therefore, 100 ns, which is 100 cycles based on 1 GHz, is designated as the transition time. Based on this setup, the design parameters are automatically changed, and an

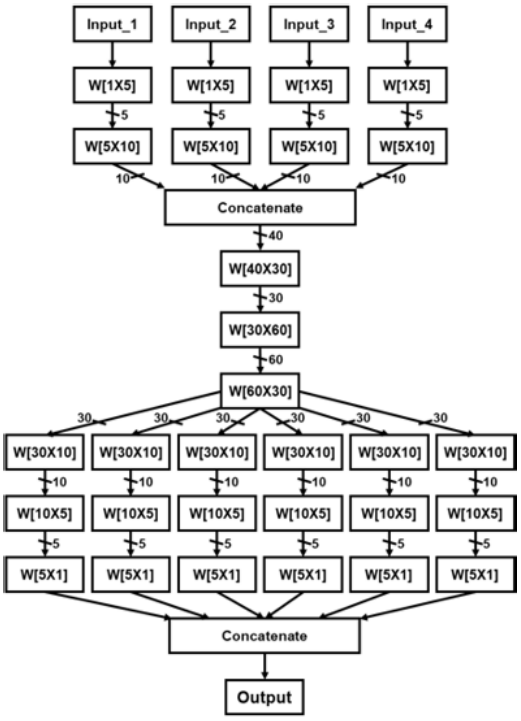


FIGURE 10. Encoder part of structure in ANN regression model employed VAE structure.

TABLE 2. Comparison of overall running time and normalized step size between multi-step and single-step simulation.

	Multi-step	Single-step
Coarse simulation	8.2 min	55.1 min
Verification (I)	20 min	-
Fine simulation	8.2 min $\times 5$	-
Verification (II)	10 min	2 min
Overall running time	79.2 min	57.1 min
Normalized step size	0.278	1

automated design is implemented using a code that can be run.

V. RESULTS

We optimized the circuit by applying all previously suggested methods. The circuit is optimized based on 45 nm, 65 nm, and 180 nm Process-Design-Kit (PDK) to prove that it is easy to migrate to other tech nodes such as digital circuits. In the verification step, the temperature is set to 120 °C in the SS condition, and the FF is set to - 40 °C for the simulation.

A. COARSE AND FINE SIMULATION RESULT

Fig. 8 illustrates the error graph for the coarse and fine simulations in CMOS 65 nm. Random jitter is related to the size of the transistor; however, it varies in value for many other reasons and is highly variable. For instance, due to the random nature, it is difficult to obtain accurate results solely through transition time simulations. As a result, although the error is larger than that of the other results, considering that units are between femto seconds and pico seconds, it does not occupy a large proportion. Excluding the random jitter, it can be seen that the error for the coarse simulation mostly exists within 30 %, and for the fine simulation, it exists within approximately 5 %. The encoder structure of the ANN model is designed as shown in Fig. 10, and the decoder structure is designed in the reverse order of the encoder. The size of the verified and trained datasets is 1225. In addition, the ratio of the sizes of the training and validation datasets is divided by 9:1.

B. FINAL RESULT

We compare the final result of applying our proposed algorithm and the SPICE simulation results. Table 1 shows the results and error rates applied to CMOS 180 nm, 65 nm, and 45 nm processes. The reason for applying the proposed algorithm to various processes is to prove that migration between processes is possible. In the case of the FoM, the difference is calculated instead of the error rate. As can be seen from this table, there is no significant difference in the error rate; therefore, it can be considered suitable for migrating to other processes.

According to the FoM performance comparison in Table 1, the final results from our proposed multi-step design flow show remarkably improved accuracy performance than that of the single-step design approach. Although the final circuit's FoM performance through the multi-step algorithm is slightly lower than that of the single-step approach, the final result of the multi-step algorithm is robust to PVT variations as it has passed the sensitivity comparing process against PVT variations.

C. COMPARE SINGLE-STEP ANN REGRESSION MODEL

We additionally train and verify the algorithm that predicts the best point only for a wide range, not the zoom-in algorithm that we propose, the method applying the VAE structure, or the method that calculates the result through verification. The total amount of training data is configured equal to the sum of the data used in this study.

Fig. 9 shows the standard deviation of the error between the prediction result using only the ANN regression model and the SPICE simulation result. We compared the results of the multi-step simulation consisting of coarse and fine simulations, which are our proposed algorithms, and single-step simulation results. In addition, we applied this algorithm to the 45 nm, 65 nm, and 180 nm processes and compared the results. In the case of random jitter, since the required dimension of a regression model is much higher than the others, the final accuracy of the random jitter will be lower than that of other performance indicators. However, the remaining results did not show significant errors. We can observe that the standard deviation of the fine simulation in the multi-step simulation is smaller than that of the coarse and single-step

TABLE 3. Comparison of final result about proposed multi-step, and multi-step without VAE structure and corner simulation.

Performance Index	Proposed Multi-step			Multi-step w/o VAE and corner sim. [3]		
	Regression Model	SPICE	Error	Regression Model	SPICE	Error
Random Jitter (ps)	0.581	0.608	4.44 %	0.664	0.822	19.22 %
Period (GHz)	0.991	0.966	2.59 %	0.995	0.989	0.61 %
Duty Cycle (%)	50.424	50.417	0.01 %	51.00	50.99	0.02 %
Power Dissipation (mW)	0.464	0.470	1.28 %	0.272	0.268	1.49 %
Rising Time (ps)	167.16	164.03	1.91 %	140	137	2.19 %
Falling Time (ps)	141.90	140.16	1.24 %	169	168	0.60 %
FoM (dB)	-248.05	-247.60	0.45 (dB)	-249.21	-247.42	1.79 (dB)
FoM (dB) @ FF	-	-248.86	1.26	-	-250.47	3.05
FoM (dB) @ SS	-	-244.14	-3.46	-	-244.29	-3.13

simulations. In most of the results, the fine simulation results have significantly smaller standard deviation values than the single-step simulation results; therefore, it is more efficient to divide the simulation steps. On the other hand, as a process criterion, the smaller the unit process, the larger the standard deviation. Although it can be predicted that the smaller the unit of the process, the lower the accuracy due to mismatches and various errors, it is considered to be an acceptable error and is sufficiently applicable to other processes.

Table 2 presents the overall running time and normalized step size for both multi-step and single-step simulations. The running time is determined by concurrently executing 10 SPICE simulations. The step size in the single-step simulation is defined as a normalized step size for comparison, while in the multi-step simulation, it is relative to the step size of the fine simulation. Comparing the normalized step size, the average value in the single-step simulation is more than three times higher than in the multi-step simulation. This discrepancy becomes more pronounced as the dataset size increases, emphasizing the importance of training the regression model with denser data for improved simulation accuracy and efficiency.

D. COMPARE MULTI-STEP ANN REGRESSION MODEL WITHOUT VAE STRUCTURE AND CORNER SIMULATION

Table 3 shows that the maximum error values in the proposed multi-step approach are lower compared to the multi-step approach without the VAE structure and corner simulation. The difference in FoM is also reduced. Additionally, the proposed algorithm prioritizes robustness against corner simulation, as evidenced by a FoM variation of 4.72 dB in the proposed multi-step approach, while the multi-step approach without the VAE structure and corner simulation exhibits a FoM variation of 6.18 dB. Consequently, the design parameters are optimized to ensure robust performance in corner simulation scenarios.

VI. CONCLUSION

In this study, we propose several methods to achieve maximum efficiency with minimal simulation data. The zoom-in algorithm divides the sweep range of the design parameters into coarse and fine simulations. In other words, the role of each step can be divided into finding and predicting the accurate points. Thus, the zoom-in algorithm makes efficient searching possible. In addition, the VAE structure detects the points at which the ANN model is mistrained. Finally, through the verification step, a design point robust to PVT variations is determined.

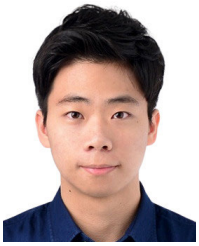
Furthermore, the fine simulation predicts the necessary points for a specific candidate group, but the ANN model created in the coarse simulation uses data computed based on the overall simulation range. Models trained with coarse simulation data have the advantage that they can be reused if other VCO performance is needed. Thus, when a VCO is required in another circuit, it is possible to immediately designate a candidate group using the created ANN Model. In addition, if the ANN model is trained with a much larger dataset, the design time is significantly reduced, and the accuracy is higher. Therefore, it has tremendous advantages when designing various specifications in the same circuit topology.

REFERENCES

[1] G. Huang, J. Hu, Y. He, J. Liu, M. Ma, Z. Shen, J. Wu, Y. Xu, H. Zhang, and K. Zhong, "Machine learning for electronic design automation: A survey," *ACM Trans. Design Autom. Electron. Syst.*, vol. 26, no. 5, pp. 1–46, Jun. 2021.

[2] Q. Zhang, J. J. Liou, J. Mcmacken, J. Thomson, and P. Layman, "Modeling of mismatch effect in submicron MOSFETs based on BSIM3 model and parametric tests," *IEEE Electron Device Lett.*, vol. 22, no. 3, pp. 133–135, Mar. 2001.

[3] J.-W. Hyun and J.-W. Nam, "Regression model-based VCO design optimization technique," in *Proc. Int. Conf. Electron., Inf., Commun. (ICEIC)*, 2023, pp. 1–3.
[4] P. M. VanNostrand, I. Kyriazis, M. Cheng, T. Guo, and R. J. Walls, "Confidential deep learning: Executing proprietary models on untrusted devices," 2019, *arXiv:1908.10730*.
[5] N. Lourenço, E. Afacan, R. Martins, F. Passos, A. Canelas, R. Póvoa, N. Horta, and G. Dundar, "Using polynomial regression and artificial neural networks for reusable analog IC sizing," in *Proc. 16th Int. Conf. Synth., Model., Anal. Simul. Methods Appl. Circuit Design (SMACD)*, Jul. 2019, pp. 13–16.
[6] S.-Y. Peng, B. A. Minch, and P. Hasler, "Analog VLSI implementation of support vector machine learning and classification," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2008, pp. 860–863.
[7] S. V. Thiruloga, V. K. Kukkala, and S. Pasricha, "TENET: Temporal CNN with attention for anomaly detection in automotive cyber-physical systems," in *Proc. 27th Asia South Pacific Design Autom. Conf. (ASP-DAC)*, Jan. 2022, pp. 326–331.
[8] S. R. Chowdhury, S. Bhardwaj, and J. Kitchen, "Design automation of CMOS Op-amps using statistical geometric programming," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2022, pp. 1575–1579.
[9] J. Nam and Y. K. Lee, "Machine-learning based analog and mixed-signal circuit design and optimization," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Jan. 2021, pp. 874–876.
[10] M. Qu and M. A. Styblinski, "An adaptive approach to statistical macromodeling of analog integrated circuits," in *Proc. Int. Symp. Circuits Syst.*, vol. 3, Sep. 1995, pp. 1596–1599.
[11] S. Su, Q. Zhang, J. Liu, M. Hassanpourghadi, R. Rasul, and M. S. Chen, "TAFA: Design automation of analog mixed-signal FIR filters using time approximation architecture," in *Proc. 27th Asia South Pacific Design Autom. Conf. (ASP-DAC)*, Jan. 2022, pp. 526–531.
[12] J.-W. Nam, Y.-K. Cho, and Y. K. Lee, "Regression model-based AMS circuit optimization technique utilizing parameterized operating condition," *Electronics*, vol. 11, no. 3, pp. 408–422, Jan. 2022.
[13] E. Maricau, D. De Jonghe, and G. Gielen, "Hierarchical analog circuit reliability analysis using multivariate nonlinear regression and active learning sample selection," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2012, pp. 745–750.
[14] M. Barros, J. Guilherme, and N. Horta, "GA-SVM optimization kernel applied to analog IC design automation," in *Proc. 13th IEEE Int. Conf. Electron., Circuits Syst.*, Dec. 2006, pp. 486–489.
[15] D. Ghai, S. P. Mohanty, and G. Thakral, "Fast analog design optimization using regression-based modeling and genetic algorithm: A nano-CMOS VCO case study," in *Proc. Int. Symp. Quality Electron. Design (ISQED)*, Mar. 2013, pp. 406–411.
[16] G. G. E. Gielen and R. A. Rutenbar, "Computer-aided design of analog and mixed-signal integrated circuits," *Proc. IEEE*, vol. 88, no. 12, pp. 1825–1854, Dec. 2000.
[17] Q. Xu, Y. Yang, Z. Wu, and L. Zhang, "Different latent variables learning in variational autoencoder," in *Proc. 4th Int. Conf. Inf., Cybern. Comput. Social Syst. (ICCSS)*, Jul. 2017, pp. 508–511.
[18] X. Gao, E. A. M. Klumperink, P. F. J. Geraedts, and B. Nauta, "Jitter analysis and a benchmarking figure-of-merit for phase-locked loops," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 56, no. 2, pp. 117–121, Feb. 2009.



JIN-WON HYUN received the B.S. degree in electronic and IT media engineering from the Seoul National University of Science and Technology, Seoul, South Korea, in 2023, where he is currently pursuing the M.S. degree in electronic engineering. His research interests include AMS circuit design automation, clock generation circuits, and low-power high-speed high-resolution analog-to-digital data converters.

JAE-WON NAM (Member, IEEE) received the B.S. and M.S. degrees from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2006 and 2008, respectively, and the Ph.D. degree in electrical engineering from the University of Southern California (USC), Los Angeles, CA, USA, in 2019. From 2008 to 2012, he was with the Electronics and Telecommunications Research Institute (ETRI), Daejeon, as a full-time Researcher. Since Fall 2017, he has been a Graduate Intern with the Data Center Group, Intel Corporation, Santa Clara, CA, USA, involved in next-generation high-speed I/O architectures. From July 2019 to July 2020, he was an Analog Engineer with the I/O Circuit Technology Team, Intel Corporation Ltd. He is currently an Assistant Professor with the Department of Electronic Engineering, Seoul National University of Science and Technology (SeoulTech). His research interests include designing low-power high-speed high-resolution analog-to-digital data converters and high-speed I/O interface circuits.