

Research Article

A Methodology for Generation of Performance Models for the Sizing of Analog High-Level Topologies

Soumya Pandit,¹ Chittaranjan Mandal,² and Amit Patra²

¹*Institute of Radio Physics and Electronics, University of Calcutta, Kolkata 700009, India*

²*Indian Institute of Technology, Kharagpur 721302, India*

Correspondence should be addressed to Soumya Pandit, soumya.pandit@yahoo.co.in

Received 4 March 2011; Accepted 28 May 2011

Academic Editor: Sheldon Tan

Copyright © 2011 Soumya Pandit et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a systematic methodology for the generation of high-level performance models for analog component blocks. The transistor sizes of the circuit-level implementations of the component blocks along with a set of geometry constraints applied over them define the sample space. A Halton sequence generator is used as a sampling algorithm. Performance data are generated by simulating each sampled circuit configuration through SPICE. Least squares support vector machine (LS-SVM) is used as a regression function. Optimal values of the model hyper parameters are determined through a grid search-based technique and a genetic algorithm- (GA-) based technique. The high-level models of the individual component blocks are combined analytically to construct the high-level model of a complete system. The constructed performance models have been used to implement a GA-based high-level topology sizing process. The advantages of the present methodology are that the constructed models are accurate with respect to real circuit-level simulation results, fast to evaluate, and have a good generalization ability. In addition, the model construction time is low and the construction process does not require any detailed knowledge of circuit design. The entire methodology has been demonstrated with a set of numerical results.

1. Introduction

An analog high-level design process is defined as the translation of analog system-level specifications into a proper topology of component blocks, in which the specifications of all the component blocks are completely determined so that the overall system meets its desired specifications optimally [1–3]. The two important steps of an analog high-level design procedure are high-level topology generation/selection [4, 5] and high-level specification translation [6]. At the high-level design abstraction, a topology is defined as an interconnection of several analog component blocks such as amplifier, mixer and filter. The detailed circuit-level implementations of these component blocks are not specified at this level of abstraction. The analog component blocks are represented by their high-level models.

During the past two decades, many optimization-based approaches have been proposed to handle the task of topology generation/selection [7–11]. These approaches involve the task of topology sizing, where the specification parameters of all the component blocks of a topology are

determined such that the desired system specifications are optimally satisfied. The two important modules for this type of design methodology are a performance estimation module and an optimization engine. The implementation of the design methodology is based upon the flow of information between these two modules.

The performance models that are used in the high-level design abstraction are referred to as high-level performance models. An analog high-level performance model is a function that estimates the performance of an analog component block when some high-level design parameters of the block are given as inputs [12, 13]. The important requirements for a good high-level performance model are as follows. (i) The model needs to be low dimensional. (ii) The predicted results need to be accurate. The model accuracy is measured as the deviation of the model predicted value from the true function value. The function value in this case is the performance parameter obtained from transistor level simulation [12]. (iii) The evaluation time must be short. This is measured by the CPU time required to evaluate a model. (iv) The time required to construct an accurate model must

be small, so that the design overhead does not become high. As a rough estimate, the construction cost is measured as

$$T_{\text{construction}} = T_{\text{data generation}} + T_{\text{training}}, \quad (1)$$

where the terms are self-explanatory. There exists a tradeoff between these requirements since a model with lower prediction error generally takes more time for construction and evaluation.

In this work, we have developed the performance models using least squares support vector machine (LS-SVM) as the regressor. The transistor sizes of the circuit-level implementations of the component blocks along with a set of geometry constraints applied over them define the sample space. Performance data are generated by simulating each sampled circuit configuration through SPICE. The LS-SVM hyper parameters are determined through formal optimization-based techniques. The constructed performance models have been used to implement a high-level topology sizing process. The advantages of this methodology are that the constructed models are accurate with respect to real circuit-level simulation results, fast to evaluate and have a good generalization ability. In addition, the model construction time is low and the construction process does not require any detailed knowledge of circuit design. The entire methodology has been demonstrated with a set of experimental results.

The rest of the paper is organized as follows. Section 2 reviews some related works. Section 3 presents the background concepts on least squares support vector machines. An outline of the methodology is provided in Section 4. The model generation methodology is described in detail in Section 5. The topology sizing process is described in Section 6. Numerical results are provided in Section 7 and finally conclusion is drawn in Section 8.

2. Related Work

A fairly complete survey of related works is given in [14]. An analog performance estimation (APE) tool for high-level synthesis of analog integrated circuits is described in [15, 16]. It takes the design parameters (e.g., transistor sizes, biasing) of an analog circuit as inputs and determines its performance parameters (e.g., power consumption, thermal noise) along with anticipated sizes of all the circuit elements. The estimator is fast to evaluate but the accuracy of the estimated results with respect to real circuit-level simulation results is not good. This is because the performance equations are based on simplified MOS models (SPICE level 1 equations). A power estimation model for ADC using empirical formulae is described in [13]. Although this is fast, the accuracy with respect to real simulation results under all conditions is off by orders of magnitude. The technique for generation of posynomial equation-based performance estimation models for analog circuits like op-amps, multistage amplifiers, switch capacitor filters, and so forth, is described in [17, 18]. An important advantage of such a modeling approach is that the topology sizing process can be formulated as a geometric program, which is easy to solve through very fast techniques. However, there are several limitations of this technique. The derivation of performance equations is often a manual

process, based on simple MOS equations. In addition, although many analog circuit characteristics can be cast in posynomial format, this is not true for all characteristics. For such characteristics, often an approximate representation is used. An automatic procedure for generation of posynomial models using fitting technique is described in [19, 20]. This technique overcomes several limitations of the handcrafted posynomial modeling techniques. The models are built from a set of data obtained through SPICE simulations. Therefore, full accuracy of SPICE simulation is achieved through such performance models. A neural network-based tool for automated power and area estimation is described in [21]. Circuit simulation results are used to train a neural network model, which is subsequently used as an estimator. Fairly recently, support vector machine (SVM) has been used for modeling of performance parameters for RF and analog circuits [22–24]. In [25], SVM-optimized by GA has been used to develop a soft fault diagnosis method for analog circuits. In [26], GA and SVM has been used in conjunction for developing feasibility model which is then used within an evolutionary computation-based optimization framework for analog circuit optimization.

2.1. Comparison with Existing Methodologies. The present methodology uses nonparametric regression technique for constructing the high-level performance models. Compared with the other modeling methodologies employing symbolic analysis technique or simulation-based technique, the advantages of the present methodology are as follows. (i) Full accuracy of SPICE simulations and advanced device models, such as BSIM3v3 are used to generate the performance models. The models are thus accurate compared to real circuit-level simulation results. (ii) There is no need for any *a priori* knowledge about the unknown dependency between the inputs and the outputs of the models to be constructed. (iii) The generalization ability of the models is high. (iv) The model construction time is low and the construction process does not require any detailed circuit design knowledge.

The EsteMate methodology [21] using artificial neural network (ANN) and the SVM-based methodology discussed in [22, 23] are closely related with the present methodology. The methodology that we have developed, however, has a number of advantages over them. These are as follows.

- (1) In the EsteMate methodology, the specification parameters of a component block constitute the sample space for training data generation. The specification parameters are electrical parameters and there exists strong nonlinear correlations amongst them. Therefore, sophisticated sampling strategies are required for constructing models with good generalization ability in the EsteMate methodology. On the other hand, in our method, the transistor sizes along with a set of geometry constraints applied over them define the sample space. Within this sample space, the circuit performance behavior becomes weakly nonlinear. Thus simple sampling strategies are used in our methodology to construct models with good generalization ability.

- (2) In EsteMate, for each sample, a complete circuit sizing task using a global optimization algorithm is required for generation of the training data. This is usually prohibitively time consuming. On the other hand, in our method, simple circuit simulations using the sampled transistor sizes are required for data generation. Therefore, the cost of training data generation in our method is much less compared to that in the EsteMate methodology [21]. With the EsteMate methodology, the training sample points are so generated that performances such as power is optimized. On the other hand, in our methodology, the task of performance optimization has been considered as a separate issue, isolated from the performance model generation procedure. Our strategy is actually followed in all practical optimization-based high-level design procedures [1, 27].
- (3) The generalization ability of the models constructed with our methodology is better than that generated through the EsteMate methodology. This is because the latter uses ANN regression technique. Neural network-based approaches suffer from difficulties with generalization, producing models that can overfit the data. This is a consequence of the optimization algorithms used for parameter selection and the statistical measures used to select the “best” model. SVM formulation, on the other hand, is based upon structural risk minimization (SRM) principle [28], which has been shown to be superior to traditional empirical risk minimization (ERM) principle, employed by the conventional neural networks. SRM minimizes an upper bound on the expected risk, as opposed to ERM that minimizes the error on the training data. Therefore an SVM has greater generalization capability.
- (4) The SVM-based methodology, as presented in [23], uses heuristic knowledge to determine the model hyper parameters. The present methodology uses optimization techniques to determine optimal values for them. GA-based methodology for determination of optimal values for the model hyper parameters is found to be faster compared to the grid search technique employed in [22].

3. Background: Least Squares Support Vector Regression

In recent years, the support vector machine (SVM), as a powerful new tool for data classification and function estimation, has been developed [28]. Suykens and Vandewalle [29] proposed a modified version of SVM called least squares SVM. In this subsection, we briefly outline the theory behind the LS-SVM as function regressor.

Consider a given set of training samples $\{x_k, y_k\}_{k=1,2,\dots,N_{tr}}$ where x_k is the input value and y_k is the corresponding target

value for the k th sample. With an SVR, the relationship between the input vector and the target vector is given as

$$\hat{y}(x) = w^T \phi(x) + b, \quad (2)$$

where ϕ is the mapping of the vector \bar{x} to some (probably high-dimensional) feature space, b is the bias, and w is the weight vector of the same dimension as the feature space. The mapping $\phi(\bar{x})$ is generally nonlinear which makes it possible to approximate nonlinear functions. The approximation error for the k th sample is defined as

$$e_k = y_k - \hat{y}_k(x_k). \quad (3)$$

The minimization of the error together with the regression is given as

$$\min \mathcal{J}(w, e) = \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{k=1}^{N_{tr}} e_k^2, \quad (4)$$

with equality constraint

$$y_k = w^T \phi(x_k) + b + e_k, \quad k = 1, 2, \dots, n, \quad (5)$$

where N_{tr} denotes the total number of training datasets and the suffix k denotes the index of the training set, that is, k th training data, γ is the regularization parameter.

The optimization problem (4) is considered to be a constrained optimization problem and a Lagrange function is used to solve it. Instead of minimizing the primary objective (4), a dual objective, the so-called Lagrangian, is formed of which the saddle point is the optimum. The Lagrangian for this problem is given as

$$\mathcal{L}(w, b, e, \alpha) = \mathcal{J}(w, e) - \sum_{k=1}^{N_{tr}} \alpha_k (w^T \phi(x_k) + b + e_k - y_k), \quad (6)$$

where α_k 's are called the Lagrangian multipliers. The saddle point is found out by setting the derivatives equal to zero:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial w} = 0 &\longrightarrow w = \sum_{k=1}^{N_{tr}} \alpha_k \phi(x_k), \\ \frac{\partial \mathcal{L}}{\partial b} = 0 &\longrightarrow w = \sum_{k=1}^{N_{tr}} \alpha_k = 0, \\ \frac{\partial \mathcal{L}}{\partial e_k} = 0 &\longrightarrow \alpha_k = \gamma e_k, \\ \frac{\partial \mathcal{L}}{\partial \alpha_k} = 0 &\longrightarrow w^T \phi(x_k) + b + e_k - y_k = 0. \end{aligned} \quad (7)$$

By eliminating e_k and w through substitution, the final model is expressed as a weighted linear combination of the inner

product between the training points and a new test object. The output is given as

$$\begin{aligned}\hat{y}(\bar{x}) &= \langle w, \phi(\bar{x}) \rangle \\ &= \left\langle \sum_{k=1}^{N_{tr}} \alpha_k \phi(x_k), \phi(x) \right\rangle + b \\ &= \sum_{k=1}^{N_{tr}} \alpha_k \langle \phi(x_k), \phi(x) \rangle + b \\ &= \sum_{k=1}^{N_{tr}} \alpha_k K(x_k, x) + b,\end{aligned}\quad (8)$$

where $K(x_k, x)$ is the kernel function. The elegance of using the kernel function lies in the fact that one can deal with feature spaces of arbitrary dimensionality without having to compute the map $\phi(\bar{x})$ explicitly. Any function that satisfies Mercer's condition can be used as the kernel function. The Gaussian kernel function used in the present work is defined as

$$K(x_k, x) = \exp\left(-\frac{\|x_k - x\|^2}{\sigma^2}\right), \quad (9)$$

and is commonly used, where σ^2 denotes the kernel bandwidth. The two important parameters, kernel parameter σ^2 , and the regulation parameter γ as defined in (4) are referred to as hyper parameters. The values of these parameters have to be determined critically in order to make the network efficient.

4. An Outline of the Methodology

The high-level performance model of an analog component block is mathematically represented as

$$\bar{p} = \bar{\mathcal{P}}(\bar{X}), \quad (10)$$

where \bar{p} is a set of performance parameters and \bar{X} is a set of specification parameters. The input specification parameters are referred to as the high-level design parameters. It is to be noted that out of various possible specification parameters, only the dominant parameters are to be considered as inputs. The selection of these is based upon the designer's knowledge [12]. These high-level design parameters describe a space referred to as the sample space. This sample space is explored to extract sample points through suitable algorithms. The numerical values of the sample points (both inputs and outputs of the performance model to be constructed) are generated through SPICE simulations. The data points so generated are divided into two sets, referred to as the training set and the test set. A least squares SVM network approximating a performance model is constructed by training the network with the training set. The test dataset is used to validate the SVM model. Suitable kernel functions are selected for constructing the SVM. An initial SVM model is constructed through some initial values of the hyper parameters. An iterative process is then executed to construct

the final LS-SVM so as to maximize its efficiency through optimal determination of the hyper parameters. An outline of the process for constructing the performance model of a single component block is illustrated in Figure 1(a).

For a complex system, consisting of many component blocks, the high-level performance model of the complete system is constructed at the second level of hierarchy, where the high-level models of the individual component blocks are combined analytically (see Figure 1(b)). The constructed performance models are used to implement a high-level topology sizing process. For a given unsized high-level topology of an analog system, the topology parameters (which are the specification parameters of the individual blocks of the high-level topology) are determined such that the desired design goals are satisfied. The entire operation is performed within an optimization procedure, which in the present work is implemented through GA. The constructed LS-SVM models are used within the GA loop. An outline of the sizing methodology is shown in Figure 1(c).

The following two important points may be noted in connection with the present methodology. First, the high-level performance model of a complete system is generated in a hierarchical manner. The major advantage of this hierarchical approach is reusability of the high-level model of the individual component blocks. The high-level model of the component blocks can be utilized whenever the corresponding component blocks are part of a system, provided the functionality and performance constraints are identical. This generally happens. The issue of reusability of the component block level high-level models is demonstrated in Experiment 3, provided later. However, this advantage comes at the cost of reduced accuracy of the model of the complete system. This tradeoff is a general phenomenon in analog design automation process. It may, however, be noted that it is possible to construct the high-level performance model of a complete system using the regression technique discussed here. For some customized applications, this may be done. Second, the requirement of low dimensionality of the models must be carefully taken care of. The scalability of our approach of model generation is not high, compared to analytical approach. However, compared to other black-box approaches like ANN-based, the scalability of our SVM-based approach is high. In addition, many of the global optimization algorithms suffer from the problem of "curse of dimensionality." For a topology sizing procedure, employing high-dimensional model the design space in which to search for optimal design points becomes too large to be handled by simple optimization algorithms. Therefore, while selecting the inputs of the model, only the dominant specification parameters need to be considered.

The detailed operations of each of the steps outlined above are discussed in the following sections and subsections.

5. High-Level Performance Model Generation

In this section, we describe the various steps of the performance model generation procedure in detail.

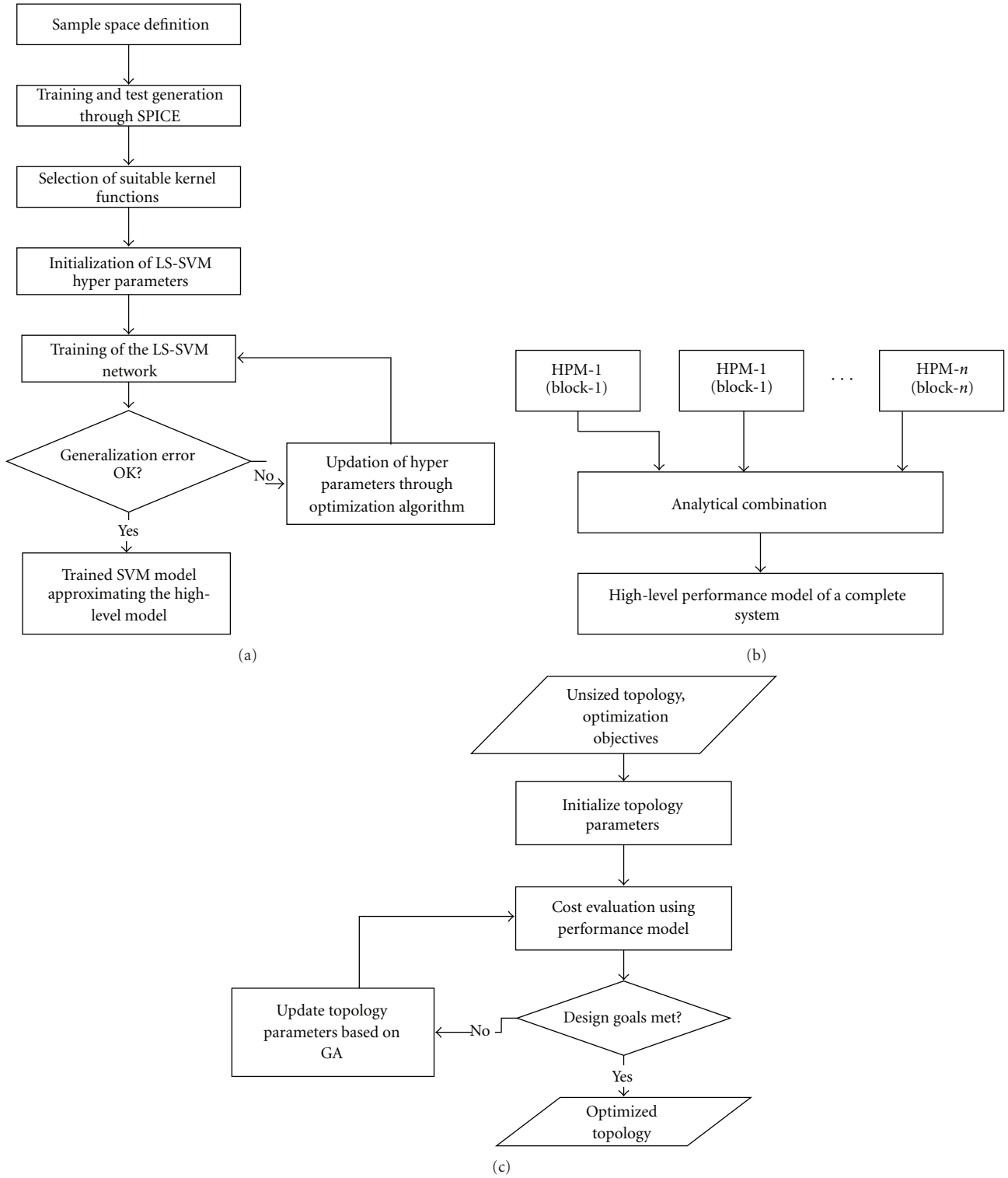


FIGURE 1: An outline of the methodology; (a) model generation for individual blocks, (b) model generation for a complete system, and (c) high-level topology sizing.

5.1. Sample Space Definition, Data Generation, and Scaling. In (10), both $\bar{\rho}$ and \bar{X} are taken to be functions of a set of geometry parameters $\bar{\alpha}$ (transistor sizes) of a component block, expressed as

$$\begin{aligned}\bar{X} &= \mathcal{R}(\bar{\alpha}), \\ \bar{\rho} &= \mathcal{Q}(\bar{\alpha}).\end{aligned}\quad (11)$$

\mathcal{R} and \mathcal{Q} represents the mapping of the geometry parameters to electrical parameters. This is illustrated in Figure 2. The multidimensional space spanned by the elements of the set $\bar{\alpha}$ is defined as circuit-level design space \mathcal{D}_α . The sample space is a subspace within \mathcal{D}_α (see Figure 3), defined through a set of geometry constraints. These geometry constraints include equality constraints as well as inequality constraints. For example, for matching purpose, the sizes of a differential pair transistors are equal. The inequality constraints are determined by the feature size of a technology and conditions that the transistors are not excessively large. With elementary algebraic transformations, all the geometry constraints are combined into a single nonlinear vector inequality, which is interpreted element wise as

$$\bar{C}_g(\bar{\alpha}) \geq 0 \iff \forall_{i \in \{1 \dots q\}} C_{gi}(\bar{\alpha}) \geq 0. \quad (12)$$

Within this sample space, the circuit performance behavior becomes weakly nonlinear [27, 30]. Therefore, simple sampling strategies are used to construct models with good generalization ability. In the present work, the sample points are extracted through Halton sequence generation. This is a quasirandom number generator which generates a set of uniformly distributed random points in the sample space [31]. This ensures a uniform and unbiased representation of the sample space. The number of sample data plays an important role in determining the efficiency of the constructed LS-SVM model. Utilizing a separate algorithm, it is possible to determine an optimum size of the training sample data such that models built with smaller training set than this optimum value will have lower accuracy than the models built with optimum number of training sample and models built with larger training data than the optimum number will have no significant higher accuracy. However, in the present work, in order to make the sampling procedure simple, the number of sample data is fixed which is determined through a trial and error method.

The training data generation process is outlined in Figure 4. For each input sample (transistor sizes) extracted from the sample space \mathcal{D}_g , the chosen circuit topology of a component block is simulated using SPICE through Cadence Spectre tool using the BSIM3v3 model. Depending upon the selected input-output parameters of an estimation function, it is necessary to construct a set of test benches that would provide sufficient data to facilitate automatic extraction of these parameters via postprocessing of SPICE output files. A set of constraints, referred to as feasibility constraints are then applied over the generated data to ensure that only feasible data are taken for training.

The generated input-output data are considered to be feasible, if either they themselves satisfy a set of constraints

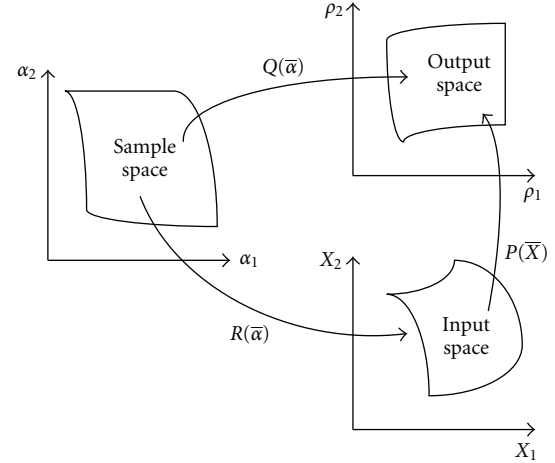


FIGURE 2: Nonlinear relation between the sample space and the input and output space.

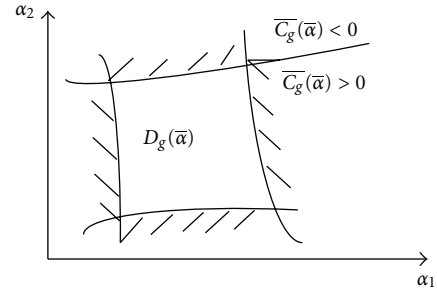


FIGURE 3: 2D projection of a four-dimensional sample space.

or the mapping procedures (\mathcal{R}, \mathcal{Q}) through which they are generated satisfy a set of constraints. The constraints are as follows [30].

- (1) Functionality constraints C_f : these constraints are applied on the measured node voltages and currents. They ensure correct functionality of the circuit and are expressed as

$$C_f = \{f_k(v, i) \geq 0, k = 1, 2, \dots, n_f\}. \quad (13)$$

For example, the transistors of a differential pair must work in saturation.

- (2) Performance constraints C_p : these are applied directly on the input-output parameters, depending upon an application system. These are expressed as

$$C_p = \{f_k(\bar{\rho}) \geq 0, f_k(\bar{X}) \geq 0, k = 1, 2, \dots, n_p\}. \quad (14)$$

For example, the phase margin of an op-amp must be greater than 45° .

The total set of constraints for feasibility checking is thus $C = \{C_f \cup C_p\}$. It is to be noted that through the process of feasibility checking, various simulation data are discarded. This at a glance may give an impression about wastage of

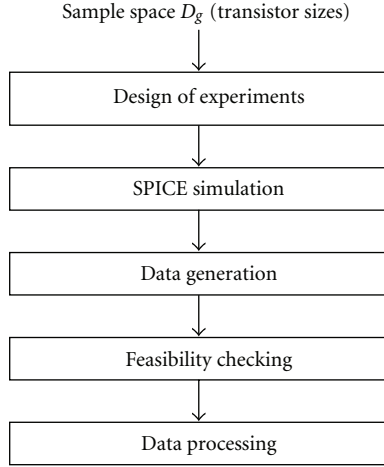


FIGURE 4: An outline of the procedure for generation of training data.

costly simulation time. However, for an analog designer (who is a user of the model), this is an important advantage. This is because, the infeasible data points will never appear as solution whenever the model is used for design characterization/optimization. Even from the model developer's perspective, this is not a serious matter considering the fact that the construction process is in general a onetime process [24]. The feasibility constraints remain invariant if the performance objectives are changed. Even if the design migrates by a small amount, these constraints usually do not change [27]. This, however, demands an efficient determination of the feasibility constraints.

Data scaling is an essential step to improve the learning/training process of SVMs. The data of the input and/or output parameters are scaled. The commonly suggested scaling schemes are linear scaling, log scaling, and two-sided log scaling. The present methodology employs both linear scaling as well as logarithmic scaling depending upon the parameters chosen. The following formula are used for linear and logarithmic scaling within an interval $[0, 1]$ [32]:

$$\begin{aligned}
 \text{Linear: } d'_j &= \frac{d_j - lb}{ub - lb}, \\
 \text{Logarithmic: } d'_j &= \frac{\log(d_j/lb)}{\log(ub/lb)},
 \end{aligned} \tag{15}$$

where d_j is the unscaled j th data of any parameter bounded within the interval $[lb, ub]$. Linear scaling of data balances the ranges of different inputs or outputs. Applying log scale to data with large variations balances large and small magnitudes of the same parameter in different regions of the model.

5.2. LS-SVM Construction. In this subsection, we discuss the various issues related to the construction of the LS-SVM regressor.

TABLE 1: List of kernel functions.

Name	Function expression
Linear kernel	$K(x_k, x) = x_k^T x$
Polynomial kernel	$K(x_k, x) = (1 + x_k^T x)^d$
RBF kernel	$K(x_k, x) = \exp\{(-\ x_k - x\ _2^2)/\sigma^2\}$
MLP kernel	$K(x_k, x) = \tanh(\kappa x_k^T x + \theta)$

5.2.1. Choice of Kernel Function. The first step of construction of an LS-SVM model is the selection of an appropriate kernel function. For the choice of kernel function $K(x_k, x)$, there are several alternatives. Some of the commonly used functions are listed in Table 1, where d , σ , κ , and θ are constants, referred to as hyper parameters. In general, in any classification or regression problem, if the hyper parameters of the model are not well selected, the predicted results will not be good enough. Optimum values for these parameters therefore need to be determined through proper tuning method. Note that the Mercer condition holds for all σ and d values in the radial basis function (RBF) and the polynomial case, but not for all possible choices of κ and θ in the multilayer perceptron (MLP) case. Therefore, the MLP kernel will not be considered in this work.

5.2.2. Tuning of Hyper Parameters. As mentioned earlier, when designing an effective LS-SVM model, the hyper parameter values have to be chosen carefully. The regularization parameter γ , determines the tradeoff cost between minimizing the training error and minimizing the model error. The kernel parameter σ or d defines the nonlinear mapping from the input space to some high-dimensional feature space [33].

Optimal values of the hyper parameters are usually determined by minimizing the estimated generalization error. The generalization error is a function that measures the generalization ability of the constructed models, that is, the ability to predict correctly the performance of an unknown sample. The techniques used for estimating the generalization error in the present methodology are as follows.

- (1) Hold-out method: this is a simple technique for estimating the generalization error. The dataset is separated into two sets, called the training set and the test set. The SVM is constructed using the training set only. Then it is tested using the test dataset. The test data are completely unknown to the estimator. The errors it makes are accumulated to give the mean test set error, which is used to evaluate the model. This method is very fast. However, its evaluation can have a high variance. The evaluation may depend heavily on the data points that end up in the training set and on those which end up in the test set, and thus the evaluation may be significantly different depending on how the division is made.
- (2) “ k ”-fold cross-validation method: in this method, the training data is randomly split into k mutually exclusive subsets (the folds) of approximately equal size [33]. The SVM is constructed using $k - 1$ of the

subsets and then tested on the subset left out. This procedure is repeated k times. Averaging the test error over the k trials gives an estimate of the expected generalization error. The advantage of this method is that the accuracy of the constructed SVM does not depend upon how the data gets divided. The variance of the resulting estimate is reduced as k is increased. The disadvantage of this method is that it is time consuming.

Primarily there are three different approaches for optimal determination of the SVM hyper parameters: heuristic method, local search method and global search method. The σ value is related to the distance between training points and the smoothness of the interpolation of the model. A heuristic rule has been discussed in [34] for estimating the σ value as $[\sigma_{\min}, \sigma_{\max}]$ where σ_{\min} is the minimum distance (non-zero) between two training points and σ_{\max} is the maximum distance between two training points. The regularization parameter γ is determined based upon the tradeoff between the smoothness of the model and its accuracy. The bigger its value the more importance is given to the error of the model in the minimization process. Choosing a low value is not suggested while using exponential RBF to model performances which are often approximately linear or weakly quadratic in most input variables. While constructing LS-SVM-based analog performance model, heuristic method has been applied for determining the hyper parameters in [23]. The hyper parameters generated through heuristic method are often found to be suboptimal as demonstrated in [12]. Therefore, determination of hyper parameters through formal optimization procedure is suggested [33].

The present methodology employs two techniques for selecting optimal values of the model hyper parameters. The first one is a grid search technique and the other one is a genetic algorithm-based technique. These are explained below considering the RBF as the kernel function. For other kernels, the techniques are accordingly used.

(1) *Grid Search Technique.* The basic steps of the grid search-based technique is outlined below.

- (1) Consider a grid space of (γ, σ^2) , defined by $\log_2 \gamma \in \{lb_\gamma, ub_\gamma\}$ and $\log_2 \sigma^2 \in \{lb_{\sigma^2}, ub_{\sigma^2}\}$, where $[lb_\gamma, ub_\gamma]$ and $[lb_{\sigma^2}, ub_{\sigma^2}]$ define the boundary of the grid space.
- (2) For each pair within the grid space, estimate the generalization error through hold-out/ k -fold cross-validation technique.
- (3) Choose the pair that leads to the lowest error.
- (4) Use the best parameter to create the SVM model as predictor.

The grid search technique is simple. However, this is computationally expensive since this is an exhaustive search technique. The accuracy and time cost of the grid method are tradeoff depending on the grid density. In general, with the increase in grid density, the computational process becomes expensive. On the other hand, sparse density lowers the

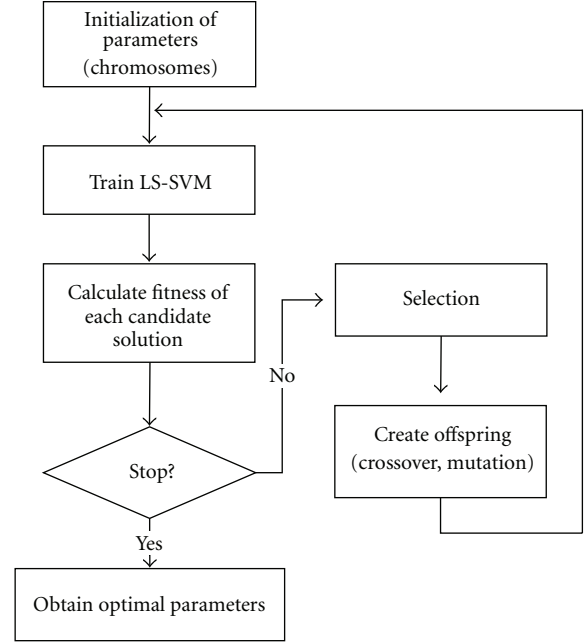


FIGURE 5: Outline of GA-based hyperparameter selection procedure.

accuracy. The grid search technique is therefore performed in two stages. In the first stage, a coarse grid search is performed. After identifying a better region on the grid, a finer grid search on that region is conducted in the second stage. In addition, the grid search process is a tricky task since a suitable sampling step varies from kernel to kernel and the grid interval may not be easy to locate without prior knowledge of the problem. In the present work, these parameters are determined through trial and error method.

(2) *Genetic Algorithm-Based Technique.* In order to reduce the computational time required to determine the optimal hyper parameter values without sacrificing the accuracy, numerical gradient-based optimization technique can be used. However, it has been found that often the SVM model selection criteria have multiple local optima with respect to the hyper parameter values [28]. In such cases, the gradient-based method have chances to be trapped in bad local optima. Considering this fact, we use a genetic algorithm-based global optimization technique for determining the hyper parameter values.

In the GA-based technique, the task of selection of the hyper parameters is same as an optima searching task, and each point in the search space represents one feasible solution (specific hyper parameters). Each feasible solution is marked by its estimated generalization ability, and the determination of a solution is equal to determination of some extreme point in the search space.

An outline of a simple GA-based process is shown in Figure 5. The chromosomes consist of two parts, $\log_2 \gamma$ and $\log_2 \sigma^2$. The encoding of the hyper parameters into a chromosome is a key issues. A realcoded scheme is used as the representation of the parameters in this work. Therefore,

the solution space coincides with the chromosome space. In order to produce the initial population, the initial values of the designed parameters are distributed in the solution space evenly. The selection of population size, is one of the factors that affects the performance of GA. The GA evaluation duration is proportional to the population size. If the population size is too large, a prohibitive amount of time for optimization will be required. On the other hand, if the population size is too small, the GA can prematurely converge to a suboptimal solution, thereby reducing the final solution quality. There is no generally accepted theory for determining optimal population size. Usually, it is determined by experimentation or experience.

During the evolutionary process of GA, a model is trained with the current hyper parameter values. The hold-out method as well as the k -fold cross-validation method are used for estimating the generalization error. The fitness function is an important factor for estimation and evolution of SVMs providing satisfactory and stable results. The fitness function expresses the users' objective and favours SVMs with satisfactory generalization ability. The fitness of the chromosomes in the present work is determined by the average relative error (ARE) calculated over the test samples. The fitness function is defined as

$$F = \frac{1}{\text{ARE}(\gamma, \sigma^2)}. \quad (16)$$

Thus, maximizing the fitness value corresponds to minimizing the predicted error. The ARE function is defined as

$$\text{ARE} = \frac{1}{N_{\text{te}} \rho'} \sum_{i=1}^{N_{\text{te}}} (\rho' - \rho). \quad (17)$$

Here N_{te} , ρ , and ρ' are the number of test data, the SVM estimator output, and the corresponding SPICE simulated value, respectively. The fitness of each chromosome is taken to be the average of five repetitions. This reduces the stochastic variability of the model training process in GA-based LS-SVM.

The genetic operator includes the three basic operators such as selection, crossover, and mutation. Roulette wheel selection technique is used for the selection operation. The probability p_i of selecting the i th solution is given by

$$p_i = \frac{F_i}{\sum_{i=1}^{N_{\text{pop}}} F_i}, \quad (18)$$

where N_{pop} is the size of the population. Besides, in order to keep the best chromosome in every generation, the idea of elitism is adopted. The use of a pair of real-parameter decision variable vectors to create a new pair of offspring vectors is done by the crossover operator. For two parent solutions \mathbf{x}_1 and \mathbf{x}_2 , the offspring is determined through a blend crossover operator. For two parent solutions \mathbf{x}_1 and \mathbf{x}_2 , such that $\mathbf{x}_1 < \mathbf{x}_2$, the blend crossover operator (BLX- β) randomly picks a solution in the range $[\mathbf{x}_1 - \beta(\mathbf{x}_2 - \mathbf{x}_1), \mathbf{x}_2 + \beta(\mathbf{x}_2 - \mathbf{x}_1)]$. Thus, if u be a random number in the range (0,1) and $\alpha = (1 + 2\beta)u - \beta$, then the following is an offspring:

$$\mathbf{x}^{\text{new}} = (1 - \alpha)\mathbf{x}_1 + \alpha\mathbf{x}_2. \quad (19)$$

If β is zero, this crossover creates a random solution in the range $(\mathbf{x}_1, \mathbf{x}_2)$. It has been reported for a number of test cases that BLX-0.5 (with $\beta = 0.5$) performs better than BLX operators with any other β value. The mutation operator is used with a low probability to alter the solutions locally to hopefully create better solutions. The need for mutation is to maintain a good diversity of the population. The normally distributed mutation operator is used in this work. A zero mean Gaussian probability distribution with standard deviation η_i for the i th solution is used. The new solution is given as

$$\mathbf{x}^{\text{new}} = \mathbf{x}_i + N(0, \eta_i). \quad (20)$$

The parameter η_i is user-defined and dependent upon the problem. Also, it must be ensured that the new solution lies within the specified upper and lower limits. When the difference between the estimated error of the child population and that of the parent population is less than a predefined threshold over certain fixed generations, the whole process is terminated and the corresponding hyper parameter pair is taken as the output.

It may be mentioned here that there is no fixed method for defining the GA parameters, which are all empirical in nature. However, the optimality of the hyper parameter values is dependent upon the values of the GA parameters. In the present work, the values of the GA parameters are selected primarily by trial and error method over several runs.

5.3. Quality Measures. Statistical functions are generally used to assess the quality of the generated estimator. The ARE function defined in (17) is one such measure. Another commonly used measure is the correlation coefficient (R). This is defined as follows:

$$R = \frac{N_{\text{te}} \sum \rho \rho' - \sum \rho \sum \rho'}{\sqrt{[N_{\text{te}} \sum \rho^2 - (\sum \rho)^2][N_{\text{te}} \sum \rho'^2 - (\sum \rho')^2]}}. \quad (21)$$

The correlation coefficient is a measure of how closely the LS-SVM outputs fit with the target values. It is a number between 0 and 1. If there is no linear relationship between the estimated values and the actual targets, then the correlation coefficient is 0. If the number is equal to 1.0, then there is a perfect fit between the targets and the outputs. Thus, the higher the correlation coefficient, the better it is.

6. Topology Sizing Methodology Using GA

The topology sizing process is defined as the task of determining the topology parameters (specification parameters of the constituent component blocks) of a high-level topology such that the desired specifications of the system are satisfied with optimized performances. In this section, we discuss a genetic algorithm-based methodology for a topology sizing process employing the constructed LS-SVM performance models.

An outline of the flow is shown in Figure 6. A high-level topology is regarded as a multidimensional space, in

TABLE 2: Transistor sizes and feasibility constraints for OTA.

	Parameters	Ranges
Transistor sizes Geometry constraints	$W_1 = W_2$	(280 nm, 400 μm)
	$W_3 = W_4 =$ $W_6 = W_7$	(1 μm , 20 μm)
	$W_8 = W_9$	(280 nm, 10 μm)
	W_5	(1 μm , 50 μm)
	$W_{10} = W_{11}$	(280 nm, 400 μm)
	C_L	(1 pF, 10 pF)
Functional constraints	Parameters	Range
	$V_{gs} - V_{th}$	≥ 0.1 V
	V_{op}	≈ 0.9 V
	V_{off}	≤ 2 mV
Performance constraints	Slew rate	(0.1 V/ μs , 20 V/ μs)
	Bandwidth	≥ 2 MHz
	DC gain	≥ 70 dB
	Phase margin	(45°, 60°)

TABLE 3: Grid search technique using hold-out method.

Model	σ^2	γ	ARE (%)		R		T_{tr}
			Training	Test	Training	Test	(min)
ρ_1	3.43	173.26	1.82	2.48	0.999	0.998	118.19
ρ_2	2.10	112.04	2.32	4.18	0.918	0.905	117.83
ρ_3	5.43	387.55	2.02	3.14	0.999	0.937	118.13

TABLE 4: Grid search technique using 5-fold cross-validation method.

Model	σ^2	γ	ARE (%)		R		T_{tr}
			Training	Test	Training	Test	(min)
ρ_1	4.10	326.32	1.27	1.33	0.999	0.999	583.12
ρ_2	2.76	112.04	2.37	2.42	0.980	0.970	583.62
ρ_3	5.33	142.65	1.82	1.85	0.998	0.998	582.67

0.05, respectively. These are determined through a trial and error process. The hyper parameter values and the quality measures are reported in Tables 5 and 6. From the results the above observations are also noted.

A comparison between the grid-search technique and the GA-based technique with respect to accuracy (ARE), correlation coefficient (R), and required training time is made in Table 7. All the experiments are performed on a PC with PIV 3.00 GHz processor and 512 MB RAM. We observe from the comparison that the accuracy of SVM models constructed using the grid search technique and the GA-based technique are almost the same. However, the GA-based technique is at least ten-times faster than the grid search method. From (1), we conclude that the construction cost of the GA-based method is much lower than the grid search-based method, since the data generation time is same for both the methods.

The scatter plots of SPICE-simulated and LS-SVM estimated values for normalized test data of the three models are shown in Figures 8(a), 8(b), and 8(c), respectively. These

TABLE 5: GA technique using hold-out method.

Model	σ^2	γ	ARE (%)		R		T_{tr}
			Training	Test	Training	Test	(min)
ρ_1	2.38	250.13	2.16	3.38	0.999	0.998	12.06
ρ_2	5.62	480.19	2.12	3.82	0.994	0.961	10.83
ρ_3	5.19	140.15	1.98	2.90	0.999	0.998	11.56

TABLE 6: GA technique using 5-fold cross-validation.

Model	σ^2	γ	ARE (%)		R		T_{tr}
			Training	Test	Training	Test	(min)
ρ_1	3.98	350.13	1.35	1.36	0.999	0.999	46.66
ρ_2	3.02	150.19	2.12	3.02	0.994	0.980	44.83
ρ_3	5.32	540.15	1.81	1.90	0.999	0.990	46.61

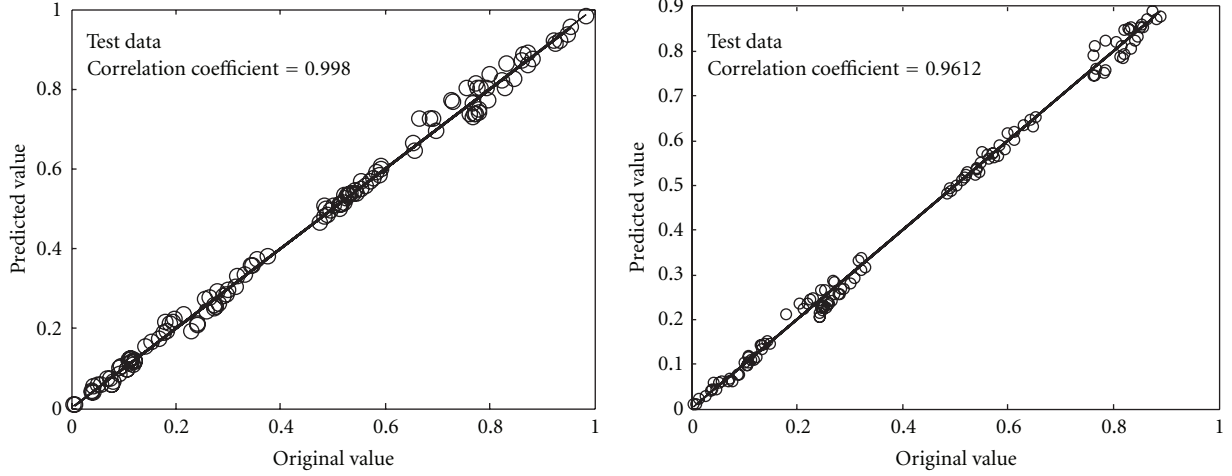
scatter plots illustrate the correlation between the SPICE simulated and the LS-SVM-estimated test data. The correlation coefficients are very close to unity. Perfect accuracy would result in the data points forming a straight line along the diagonal axis.

7.2. Experiment 2. The objective of this experimentation is to quantitatively compare between our methodology and the EsteMate [21]. The power consumption model is reconstructed using the EsteMate technique. The specification parameter space is sampled randomly. A set of 5000 samples is considered. For each selected sample, an optimal sizing is performed and the resulting power consumption is measured. The sizing is done with a simulated annealing-based optimization procedure and standard analytical equations relating transistor sizes to the specification parameters [36] following the EsteMate procedure. Of these, 3205 samples are accepted and the rest are rejected. The determination of the training set took 10 hours of CPU time. The training is done through an artificial neural network structure with two hidden layers. The number of neurons for the first layer is 9, the number of neurons for the second layer is 6. The hold-out method is used for estimating the generalization ability.

A comparison between the two methodologies is reported in Table 8. From the results, we find that the data generation time is much less in our method compared to the EsteMate method. In addition, we find that the accuracy of our method is better than the EsteMate method. The experimental observations verify the theoretical arguments given in Section 2.1.

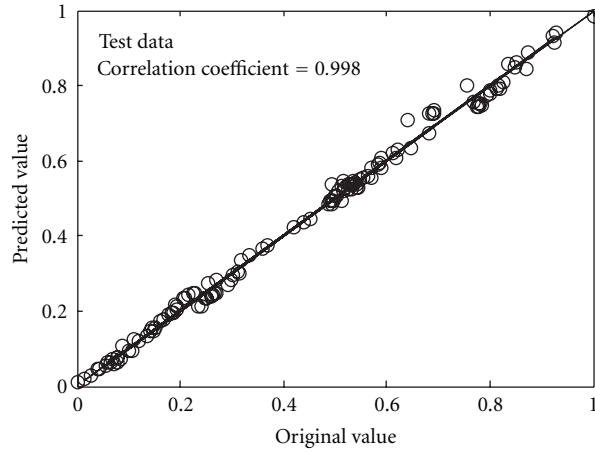
7.3. Experiment 3. The objective of this experimentation is to demonstrate the process of constructing high-level performance model of a complete system and the task of topology sizing.

System Considerations. We choose a complete analog system—interface electronics for MEMS capacitive sensor system as shown in Figure 9(a). In this configuration, a half-bridge consisting of the sense capacitors C_1, C_2 is formed and driven by two pulse signals with 180° phase difference. The amplitude of the bridge output V_x is proportional to the



(a) Scatter plot of estimated and original values for the noise model with normalized test data

(b) Scatter plot of estimated and original values for the power model with normalized test data



(c) Scatter plot of estimated and original values for the impedance model with normalized test data

FIGURE 8: Scatter plot of the constructed models.

TABLE 7: Comparison between GA and Grid search technique for LS-SVM construction.

Model	Algorithm	σ^2	γ	ARE (%)		R		T_{tr} (min)
				Training	Test	Training	Test	
ρ_1	GA	2.38	250.13	2.16	3.38	0.999	0.998	12.06
	Grid search	3.43	173.26	1.82	2.48	0.999	0.998	118.19
ρ_2	GA	5.62	480.19	2.12	3.82	0.994	0.961	10.83
	Grid search	2.10	112.04	2.32	4.18	0.980	0.905	117.83
ρ_3	GA	5.19	140.15	1.98	2.90	0.999	0.998	11.56
	Grid search	5.43	387.55	2.02	3.14	0.999	0.937	118.13

capacitance change ΔC and is amplified by a voltage amplifier. The final output voltage V_{out} is given by

$$V_{out} = V_0 \frac{2\Delta C}{2C_0 + C_p} A_v, \quad (22)$$

where C_0 is the nominal capacitance value, C_p is the parasitic capacitance value at the sensor node, V_0 is the amplitude of the applied ac signal, and A_v is the gain of the system,

depending upon the desired output voltage sensitivity. The topology employs a chopper modulation technique for low $1/f$ noise purpose.

The desired functional specifications to be satisfied are (i) output voltage sensitivity (i.e., the total gain, since the input sensitivity is known) and (ii) cutoff frequency of the filter. The performance parameters to be optimized are (i) input-referred thermal noise, (ii) total power consumption,

TABLE 8: Comparison between our methodology and EsteMate.

Method	Number of samples		ARE (%)		Generation time	Train time
	Training	Test	Training	Test		
Ours	821	206	2.12	3.82	14 min	10.83 min
EsteMate [21]	2564	641	2.88	6.53	10 hour	21 min

TABLE 9: Functional specs and design constraints.

Parameters	Desired specs
Sensing capacitance	100 fF
Capacitance sensitivity	0.4 fF
Linear range	± 6 g
Modulation frequency	1 MHz
Modulation voltage	500 mV
Input voltage sensitivity	≥ 1 mV/g
Output voltage sensitivity	≥ 100 mV/g
Cutoff frequency	≤ 40 KHz

TABLE 10: Transistor sizes and feasibility constraints for preamplifier.

Transistor sizes	Geometry constraints	
	Gm_1	Gm_2
$W_1 = W_2$	(280 nm, 400 μ m)	(280 nm, 200 μ m)
$W_3 = W_4 = W_6 = W_7$	(1 μ m, 20 μ m)	(1 μ m, 20 μ m)
$W_8 = W_9$	(280 nm, 10 μ m)	(280 nm, 10 μ m)
I_{bias}	(1 μ A, 40 μ A)	(1 μ A, 10 μ A)
Functional constraints	Parameters	Range
	$V_{gs} - V_{th}$	≥ 0.1 V
	V_{op}	≈ 0.9 V
	V_{off}	≤ 2 mV
Performance constraints	Input linearity	≥ 15 mV
	Swing	≥ 750 mV
	Bandwidth	≥ 2 MHz
	Phase margin	(45°, 60°)

and (iii) parasitic capacitance at the sensor node V_x . The functional specifications and design constraints for the system are based on [37] and are listed in Table 9.

Identification of the Component Blocks and the Corresponding Performance Models. The synthesizable component blocks are the preamplifier (PA), inverter (IN) of the phase demodulator, low-pass filter (LF), and the output amplifier (OA). These are constructed using OTAs and capacitors. Figure 9(b) shows the implementations of the amplifier and the filter blocks using OTAs and capacitor [38, 39].

High-level performance models for the synthesizable component blocks corresponding to the performance parameters—(i) input referred thermal noise, (ii) power consumption, and (iii) sensor node parasitics are constructed. The specification parameters which have dominant influence on the first two performances as well as on the functional specification, that is, the output voltage sensitivity

TABLE 11: Accuracy of preamplifier block.

Models	σ^2	γ	Training		Test	
			ARE	R	ARE	R
Noise	2.88	288.93	1.25	0.9991	1.75	0.9991
Power	1.18	203.18	2.05	0.9989	2.35	0.9989
Parasitics	3.45	123.93	0.58	0.9999	0.62	0.9999

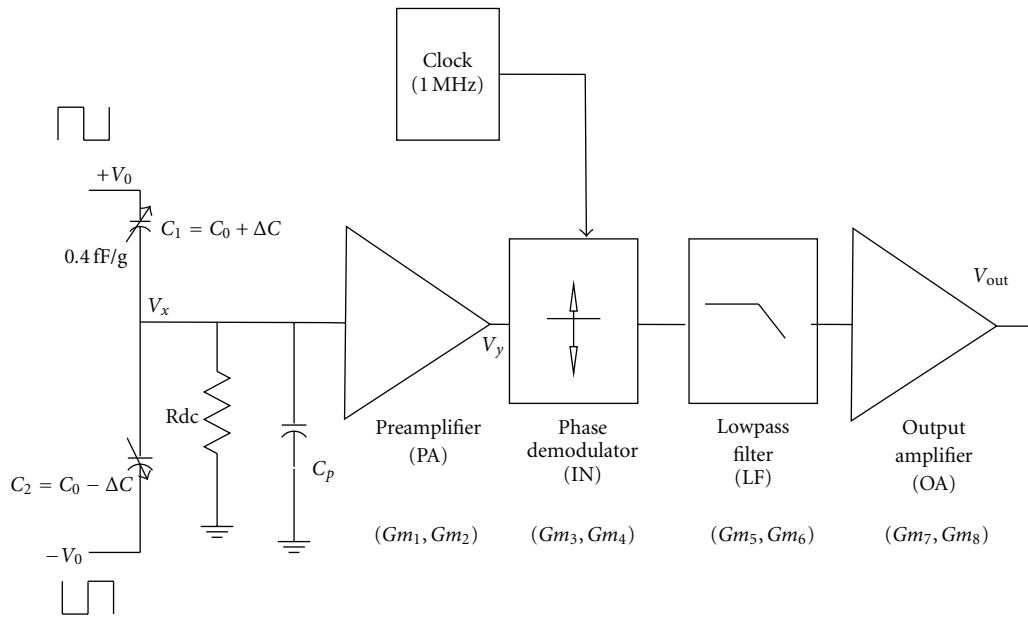
and the cutoff frequency are the transconductance values of all the OTAs involved. On the other hand, for the last performance parameter, that is, sensor node parasitics, transconductance value of the first OTA of the preamplifier block is the single design parameter. Thus the Gm values of the OTAs are considered as high-level design parameters. In summary, we construct three performance models, input referred thermal noise, power consumption, and sensor node parasitics as functions of the Gm values of the OTAs.

Construction of Performance Models for the PA Block. The geometry constraints and the feasibility constraints for the PA block of the topology are tabulated in Table 10. Similar types of constraints are considered for the other component blocks also. The input-output parameters of the models to be constructed are extracted through techniques discussed earlier. The sensor node parasitic capacitance is measured utilizing the half-bridge circuit shown in Figure 9(a), with only one amplifier block. Considering $\Delta C = 5$ fF, $C_0 = 65$ fF, a square wave signal with amplitude $V_0 = 500$ mV is applied and transient analysis is performed. Measuring the signal at the node V_x , C_p is calculated using (22).

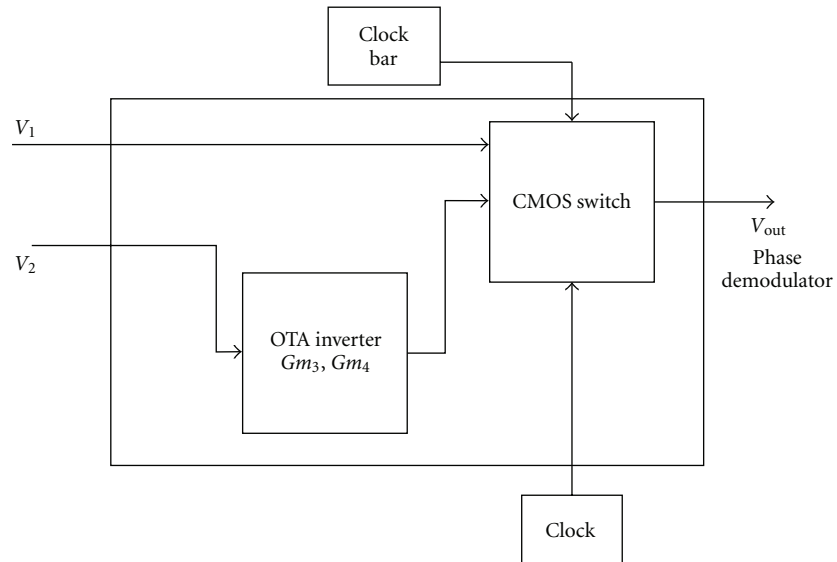
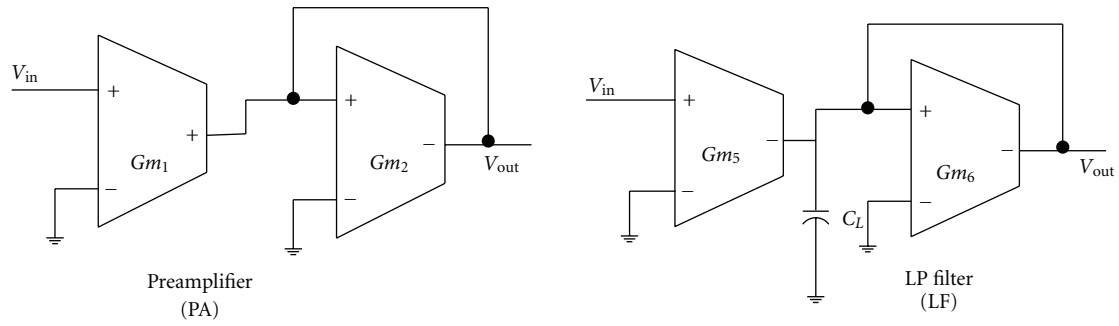
Table 11 shows the hyper parameter values, percentage average relative error, and correlation coefficient of the constructed performance models for the preamplifier, with respect to SPICE-simulated value.

Reusability of Models and Construction of High-Level Model for the Complete System. The performance models corresponding to the noise and the power consumption for the PA block are reused for the other component blocks. This is because all the component blocks have topological similarities and each of them is constructed from OTA circuits, as demonstrated in Figure 9(b). The issue of reusability of individual high-level models in a complete system is thus applied here.

The high-level models of the PA, IN, LF and OA blocks are combined analytically to construct the model of the complete system. The input referred noise and power



(a) Voltage sensing configuration of the interface electronics for MEMS capacitive sensor



(b) OTA-C realizations of amplifier and filter

FIGURE 9: Considered system for experiment 3.

TABLE 12: Synthesized topology parameters.

Topology parameters	Synthesized value
Gm_1	216.30 μS
Gm_2	14.67 μS
Gm_3	17.97 μS
Gm_4	16.80 μS
Gm_5	15.92 μS
Gm_6	13.96 μS
Gm_7	131.73 μS
Gm_8	16.15 μS
C_L	63 pF

consumption of the total system is given by

$$V_{nT}^2 = V_{n1}^2(Gm_1, Gm_2) + \frac{V_{n2}^2(Gm_3, Gm_4)}{A_1^2} + \frac{V_{n3}^2(Gm_5, Gm_6)}{A_1^2} + \frac{V_{n4}^2(Gm_7, Gm_8)}{A_1^2}, \quad (23)$$

$$P_T = P_1(Gm_1, Gm_2) + P_2(Gm_1, Gm_2) + P_3(Gm_1, Gm_2) + P_4(Gm_1, Gm_2). \quad (24)$$

A_1 is the gain of the preamplifier. $V_{n1}(Gm_1, Gm_2)$ is the thermal noise model for the PA block, $V_{n2}(Gm_3, Gm_4)$ is that for the IN block of the phase demodulator, and so on. It is to be noted that $V_{n2}(Gm_3, Gm_4)$ need not be constructed again. It is same as $V_{n1}(Gm_1, Gm_2)$. This is true for $V_{n3}(Gm_5, Gm_6)$ and $V_{n4}(Gm_7, Gm_8)$. This reusability principle is applied for the power consumption model of all the blocks. The sensor node parasitics $P_a = P_a(Gm_1)$ is the same as the input parasitics of the preamplifier. It is to be noted that while constructing the high-level performance model of a complete system, the interactions between the transistors are taken care of while constructing the component-level performance model utilizing SPICE simulation data and the coupling between the blocks are considered through analytical equations.

Optimization Problem Formulation and Results. With these, the optimization problem for the topology sizing task is formulated as follows:

$$\text{Minimize } \omega_1 V_{nT} + \omega_2 P_T + \omega_3 P_a$$

$$\text{such that } (V_{out})_{target} - V_{in} \left[\frac{Gm_1}{Gm_2} \frac{Gm_3}{Gm_4} \frac{Gm_5}{Gm_6} \frac{Gm_7}{Gm_8} \right] \leq \epsilon_1$$

$$f_c - \frac{Gm_6}{2\pi C_L} \leq \epsilon_2$$

$$Gm_{i\min} \leq Gm_i \leq Gm_{i\max}$$

$$C_{L\min} \leq C_L \leq C_{L\max},$$

(25)

where ω_i are the associated weights.

The target output voltage sensitivity of the system (i.e., the total gain of the system) is taken as 145 mV/g and the

TABLE 13: Comparison of predicted performances and SPICE value.

Performances	Pred	SPICE	Error %
Noise (nV/ \sqrt{Hz})	19.65	20.32	3.40
Power (μW)	572.78	592	3.36
Parasitics (fF)	92.05	94.12	2.24
Sensitivity	145.16	138	4.93
Cutoff (KHz)	35.28	38	7.70

cutoff frequency is taken as 35 KHz. The synthesis procedure took 181 seconds on a PIV, 3.00 GHz processor PC with 512 MB RAM. The crossover and the mutation probability are taken as 0.85 and 0.05, respectively. These are determined through a trial and error process. Table 12 lists the synthesized values of the topology parameters, as obtained from the synthesis procedure.

Validation. To validate the synthesis procedure, we simulate the entire system at the circuit-level using SPICE. Exact values of Gm are not achievable often. In such cases, the nearest neighbouring values are realized. An approximate idea about the transistor sizes required to implement the synthesized Gm values are made from the large set of data gathered during the estimator construction. A comparison between the predicted performances and simulated values is presented in Table 13. We observe that the relative error between predicted performances and simulated performances in each case is acceptable. However, for the output sensitivity and the cutoff frequency, the error is high. This is because the circuit-level nonideal effects have not been considered in the topology sizing process while formulating the final cost function and constraint functions. Following conventional procedure, this has been done purposefully in order to make the functions simple and the process converge smoothly [1, 27]. The acceptability and feasibility of the results are ensured to a large extent, since the utilized model is based on SPICE simulation results. The robustness of the results, however, could be verified by process corner analysis [27].

8. Conclusion

This paper presents a methodology for generation of high-level performance models for analog component blocks using nonparametric regression technique. The transistor sizes of the circuit-level implementations of the component blocks along with a set of geometry constraints applied over them define the sample space. Performance data are generated by simulating each sampled circuit configuration through SPICE. Least squares support vector machine (LS-SVM) is used as a regression function. The generalization ability of the constructed models has been estimated through a hold-out method and a 5-fold cross-validation method. Optimal values of the model hyper parameters are determined through a grid search-based technique and a GA-based technique. The high-level models of the individual component blocks are combined analytically to construct the high-level model of a complete system. The entire methodology has been implemented under MATLAB environment.

The methodology has been demonstrated with a set of experiments. The advantages of the present methodology are that the constructed models are accurate with respect to real circuit-level simulation results, fast to evaluate and have a good generalization ability. In addition, the model construction time is low and the construction process does not require any detailed knowledge of circuit design. The constructed performance models have been used to implement a GA-based topology sizing process. The process has been demonstrated by considering the interface electronics for an MEMS capacitive accelerometer sensor as an example. It may be noted that multiobjective optimization algorithms [40] can also be used in the proposed approach for solving (25).

Acknowledgment

The first author likes to thank the Department of Science and Technology, Government of India for partial financial support of the present paper through Fast Track Young Scientist Scheme, no. SR/FTP/ETA-063/2009.

References

- [1] E. S. J. Martens and G. G. E. Gielen, *High-Level Modeling and Synthesis of Analog Integrated Systems*, Springer, New York, NY, USA, 2008.
- [2] G. G. E. Gielen, "CAD tools for embedded analogue circuits in mixed-signal integrated systems on chip," *IEE Proceedings: Computers and Digital Techniques*, vol. 152, no. 3, pp. 317–332, 2005.
- [3] S. Y. Lee, C. Y. Chen, J. H. Hong, R. G. Chang, and M. P.-H. Lin, "Automated synthesis of discrete-time sigma-delta modulators from system architecture to circuit netlist," *Microelectronics Journal*, vol. 42, pp. 347–357, 2010.
- [4] E. Martens and G. G. E. Gielen, "Classification of analog synthesis tools based on their architecture selection mechanisms," *Integration, the VLSI Journal*, vol. 41, no. 2, pp. 238–252, 2008.
- [5] S. Pandit, C. Mandal, and A. Patra, "An automated high-level topology generation procedure for continuous-time $\Sigma\Delta$ modulator," *Integration, the VLSI Journal*, vol. 43, no. 3, pp. 289–304, 2010.
- [6] S. Pandit, S. K. Bhattacharya, C. Mandal, and A. Patra, "A fast exploration procedure for analog high-level specification translation," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 8, pp. 1493–1497, 2008.
- [7] G. G. E. Gielen, "Modeling and analysis techniques for system-level architectural design of telecom front-ends," *IEEE Transactions on Microwave Theory and Techniques*, vol. 50, no. 1, pp. 360–368, 2002.
- [8] J. Crols, S. Donnay, M. Steyaert, and G. G. E. Gielen, "High-level design and optimization tool for analog RF receiver front-ends," in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD '95)*, pp. 550–553, November 1995.
- [9] F. Medeiro, B. Perez-Verdu, A. Rodriguez-Vazquez, and J. L. Huertas, "Vertically integrated tool for automated design of $\Sigma\Delta$ modulators," *IEEE Journal of Solid-State Circuits*, vol. 30, no. 7, pp. 762–772, 1995.
- [10] H. Tang and A. Doboli, "High-level synthesis of $\Delta\Sigma$ modulator topologies optimized for complexity, sensitivity, and power consumption," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 25, no. 3, pp. 597–607, 2006.
- [11] Y. Wei, A. Doboli, and H. Tang, "Systematic methodology for designing reconfigurable $\Delta\Sigma$ modulator topologies for multimode communication systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 3, pp. 480–495, 2007.
- [12] S. Pandit, C. Mandal, and A. Patra, "Systematic methodology for high-level performance modeling of analog systems," in *Proceedings of the 22nd International Conference on VLSI Design—Held Jointly with the 7th International Conference on Embedded Systems (VLSID '09)*, pp. 361–366, January 2009.
- [13] E. Lauwers and G. G. E. Gielen, "Power estimation methods for analog circuits for architectural exploration of integrated systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 10, no. 2, pp. 155–162, 2002.
- [14] R. A. Rutenbar, G. G. E. Gielen, and J. Roychowdhury, "Hierarchical modeling, optimization, and synthesis for system-level analog and RF designs," *Proceedings of the IEEE*, vol. 95, no. 3, pp. 640–669, 2007.
- [15] A. Nunez-Aldana and R. Vemuri, "An analog performance estimator for improving the effectiveness of CMOS analog systems circuit synthesis," in *Proceedings of Design, Automation and Test in Europe Conference and Exhibition (DATE '99)*, pp. 406–411, Munich, Germany, 1999.
- [16] A. Doboli, N. Dhanwada, A. Nunez-Aldana, and R. Vemuri, "A two-layer library-based approach to synthesis of analog systems from VHDL-AMS specifications," *ACM Transactions on Design Automation of Electronic Systems*, vol. 9, no. 2, pp. 238–271, 2004.
- [17] M. Del Mar Hershenson, S. P. Boyd, and T. H. Lee, "Optimal design of a CMOS op-amp via geometric programming," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 20, no. 1, pp. 1–21, 2001.
- [18] P. Mandal and V. Visvanathan, "CMOS op-amp sizing using a geometric programming formulation," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 20, no. 1, pp. 22–38, 2001.
- [19] W. Daems, G. G. E. Gielen, and W. Sansen, "Simulation-based generation of posynomial performance models for the sizing of analog integrated circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, no. 5, pp. 517–534, 2003.
- [20] X. Li, P. Gopalakrishnan, Y. Xu, and L. T. Pileggi, "Robust analog/RF circuit design with projection-based performance modeling," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 26, no. 1, pp. 2–15, 2007.
- [21] G. van der Plas, J. Vandenbussche, G. G. E. Gielen, and W. Sansen, "EsteMate: a tool for automated power and area estimation in analog top-down design and synthesis," in *Proceedings of the IEEE Custom Integrated Circuits Conference (CICC '97)*, pp. 139–142, May 1997.
- [22] X. Ren and T. Kazmierski, "Performance modelling and optimisation of RF circuits using support vector machines," in *Proceedings of the 14th International Conference on Mixed Design of Integrated Circuits and Systems (MIXDES '07)*, pp. 317–321, June 2007.
- [23] T. Kiely and G. G. E. Gielen, "Performance modeling of analog integrated circuits using least-squares support vector machines," in *Proceedings of Design, Automation and Test in*

- Europe Conference and Exhibition (DATE '04)*, pp. 448–453, February 2004.
- [24] M. Ding and R. Vemuri, “A combined feasibility and performance macromodel for analog circuits,” in *Proceedings of the 42nd Design Automation Conference (DAC '05)*, pp. 63–68, June 2005.
 - [25] H. Li and Y. Zhang, “An algorithm of soft fault diagnosis for analog circuit based on the optimized SVM by GA,” in *Proceedings of the 9th International Conference on Electronic Measurement and Instruments (ICEMI '09)*, pp. 41023–41027, August 2009.
 - [26] M. Barros, J. Guilherme, and N. Horta, “Analog circuits optimization based on evolutionary computation techniques,” *Integration, the VLSI Journal*, vol. 43, no. 1, pp. 136–155, 2010.
 - [27] H. E. Graeb, *Analog Design Centering and Sizing*, Springer, New York, NY, USA, 2007.
 - [28] V. Vapnik, *Statistical Learning Theory*, Springer, New York, NY, USA, 1998.
 - [29] J. A. K. Suykens, T. V. Gestel, J. D. Brabanter, B. D. Moor, and V. J. Vandewalle, *Least Squares Support Vector Machines*, World Scientific, 2002.
 - [30] H. Graeb, S. Zizala, J. Eckmueller, and K. Antreich, “The sizing rules method for analog integrated circuit design,” in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD '01)*, pp. 343–349, San Jose, Calif, USA, November 2001.
 - [31] G. P. Box, W. G. Hunter, and J. S. Hunter, *Statistics for Experimenters: An Introduction to Design, Analysis and Model Building*, Wiley, New York, NY, USA, 1978.
 - [32] Q. J. Zhang, K. C. Gupta, and V. K. Devabhaktuni, “Artificial neural networks for RF and microwave design—from theory to practice,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 51, no. 4, pp. 1339–1350, 2003.
 - [33] K. Duan, S. S. Keerthi, and A. N. Poo, “Evaluation of simple performance measures for tuning SVM hyperparameters,” *Neurocomputing*, vol. 51, pp. 41–59, 2003.
 - [34] G. Rubio, H. Pomares, I. Rojas, and L. J. Herrera, “A heuristic method for parameter selection in LS-SVM: application to time series prediction,” *International Journal of Forecasting*, vol. 27, pp. 725–739, 2010.
 - [35] “LS-SVM Toolbox,” February 2003, <http://www.esat.kuleuven.ac.be/sista/lssvmlab/>.
 - [36] P. E. Allen and D. R. Holberg, *CMOS Analog Circuit Design*, Oxford University Press, 2004.
 - [37] J. Wu, G. K. Feeder, and L. R. Carley, “A low-noise low-offset capacitive sensing amplifier for a $50\text{-}\mu\text{g}/\sqrt{\text{Hz}}$ monolithic CMOS MEMS accelerometer,” *IEEE Journal of Solid-State Circuits*, vol. 39, no. 5, pp. 722–730, 2004.
 - [38] E. Sánchez-Sinencio and J. Silva-Martínez, “CMOS transconductance amplifiers, architectures and active filters: a tutorial,” *IEE Proceedings: Circuits, Devices and Systems*, vol. 147, no. 1, pp. 3–12, 2000.
 - [39] R. Schaumann and M. E. Van Valkenburg, *Design of Analog Filters*, Oxford University Press, 2004.
 - [40] I. Guerra-Gómez, E. Tlelo-Cuautle, T. McConaghy et al., “Sizing mixed-mode circuits by multi-objective evolutionary algorithms,” in *Proceedings of the 53rd IEEE International Midwest Symposium on Circuits and Systems (MWSCAS '10)*, pp. 813–816, August 2010.

