A SVM Surrogate Model-Based Method for Parametric Yield Optimization

Angelo Ciccazzo, Gianni Di Pillo, and Vittorio Latorre

***Abstract*—Yield optimization is a challenging topic in electronic circuit design. Methods for yield optimization based on Monte Carlo (MC) analysis of a circuit whose behavior is reproduced by simulations usually require too many time expensive simulations to be effective for iterative optimization. In this paper, we propose a method which tackles the yield optimization problem by combining a support vector machine (SVM) surrogate model (SM) of the circuit to perform the MC analysis and evaluate the yield, and an efficient optimization method to maximize the yield evaluated using the SVM SM. We report the numerical results obtained for the design of two real consumer circuits provided by STMicroelectronics, and we compare these results with the ones obtained using the industrial benchmark currently adopted at STMicroelectronics for yield optimization. These preliminary results show that the method is promising to be very efficient and capable of reaching design solutions with high values of the yield.**

***Index Terms*—Derivative-free optimization algorithm, electronic circuit design, support vector machine (SVM) surrogate model (SM), yield optimization.**

I. Introduction

We propose a methodology to maximize the yield in the electronic circuit design process. The behavior of a circuit is generally described by its performances $f_i, i = 1, \dots, m$ such as the gain, the delay between two waveforms, the phase margin, the dissipated power, and so on. For a circuit to be in full working order, all the m performances f_i must satisfy certain specifications that are generally given in terms of lower and upper bounds

$$l_i \le f_i(x_d, x_o, x_p) \le u_i \quad i = 1, \dots, m \tag{1}$$

where the variables are listed as follows.

1) x_d *(Design Variables):* These variables represent the geometrical dimensions of the components in the circuits (e.g., channel widths and lengths).
2) x_o *(Operating Variables):* These variables model operating and environmental conditions (e.g., supply voltage and temperature).
3) x_p *(Process Variables):* These variables are usually subject to uncertainty due to fluctuations in the manufacturing process and are generally modeled by Gaussian or uniform distributions (e.g., oxide thickness, threshold voltage, and channel length reduction).

We denote by $\mathcal{A}$ the feasible set of the design, operating, and process variables

$$\mathcal{A} = \big\{(x_d, x_o, x_p) \mid l_i \le f_i(x_d, x_o, x_p) \le u_i, \quad i = 1, \dots, m\big\}. \tag{2}$$

Manuscript received February 2, 2015; revised July 6, 2015 and September 18, 2015; accepted October 23, 2015. Date of publication November 16, 2015; date of current version June 16, 2016. This work was supported by the European Union through the MODERN Project under Grant ENIAC120003. This paper was recommended by Associate Editor L.-C. Wang.

A. Ciccazzo is with STMicroelectronics, Catania 95121, Italy (e-mail: angelo.ciccazzo@st.com).

G. Di Pillo and V. Latorre are with the Department of Computer, Control and Management Engineering, Sapienza University of Rome, Rome 00185, Italy (e-mail: dipillo@dis.uniroma1.it; latorre@dis.uniroma1.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCAD.2015.2501307

The yield represents the probability of a circuit to be in full working order for a certain design choice x_d, subject to given operating variables x_o and taking account of the process and environmental variations x_p.

In recent years, there is an increasing interest in yield optimization due to both the increasing number of components in a single circuit and the decrease of its size. Handling these conflicting trends is becoming more and more difficult, because they enhance the sensitivity of the circuit performances to the statistical variations in the manufacturing process. Some major challenges come from the need of making the final design robust toward these variations. This implies the use of accurate computer-aided design processes with numerical computer simulations employed in order to evaluate the circuit performances. Simulations are generally very time expensive and this reason prompts the interest in developing methods capable of performing reliable analysis with the use of less simulations as possible.

Various optimization methods have been applied to the problem of robust circuit design, see [1]. Focusing on yield optimization, two classes of methods are mainly adopted: 1) the geometric yield optimization and 2) the statistical yield optimization [2].

The geometric yield optimization [3], [4] is based on the definition of worst case distances of each performance from its lower/upper bound. The optimal design parameters are those which maximize the worst case distances of the performances. Therefore, we are led to a max min optimization problem, where the minimization determines the worst case distances. The main drawbacks of this approach are as follows.

1) The worst case distances must be computed for every performance in every design choice. This gives raise to a complex multiobjective optimization problem whose solution is expensive to obtain.
2) The calculation of the worst case distance generally is a nonlinear optimization problem with several local solutions. Therefore, the method has to be run repeatedly in order to know the best yield value that can be achieved.

Methods in this class are popular in yield optimization and are employed by WiCkeD [5], a suite for circuit analysis, modeling, sizing, and optimization. WiCkeD is the result of the research performed in [2]–[4] and is widely used in the electronic circuit industry for these kinds of applications. The WiCkeD optimization tool is based on a sequential quadratic programming method, where information on the derivatives are provided by finite differences.

The statistical yield optimization uses a Monte Carlo (MC) analysis in order to evaluate the yield, and has the advantages of greater generality and higher accuracy. However, the large number of simulations required to obtain an accurate measure of the yield makes the use of MC analysis very expensive in iterative optimization. Therefore, the efforts in literature are focused on decreasing the number of simulations, and are based on two principal strategies.

1) *Surrogate Models-Based Methods:* These methods [6] create macro-models for the yield over the design, operating, and process variables. These strategies enable the practitioners to explore the design alternatives with little computational effort, but such models suffer from a tradeoff between the number of simulations employed for the model training and its accuracy.

0278-0070 © 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

Authorized licensed use limited to: Universita degli Studi di Roma La Sapienza. Downloaded on July 29,2022 at 12:10:25 UTC from IEEE Xplore. Restrictions apply.

2) *Improved MC-Based Methods:* These methods employ alternative methods to perform the MC analysis with less simulations, but without losing information on the yield. Some are based on the latin hypercube sampling (LHS) [7] or on the quasi-MC method [8], or on computational intelligence techniques [9].

The approach proposed in this paper is a statistical yield optimization methodology which takes inspiration from both the surrogate models (SMs)-based methods and the improved MC-based methods. As a matter of facts we combine the following.

1) An accurate SM, the support vector machine (SVM) [10], to generate a reliable MC analysis and evaluate the yield.
2) An efficient DFO method [11] with fast and reliable convergence properties to maximize the yield.

In our methodology the SVM only models the process variabilities. Indeed, we embed the training of the SVM in the iterations of the optimization algorithm: at every iteration the DFO algorithm selects suitable design parameters, then an SVM is trained that only handles the information on the process parameters. This SVM is then used to generate an MC analysis with a large number of points to calculate the objective function for the DFO algorithm. Therefore, our main contributions are as follows.

1) The adoption of an efficient derivative free algorithm for circuit yield optimization that, instead of using macro models over the design, operating, and process variabilities to perform the MC yield analysis, generates surrogates models only in the design points of actual interest for the algorithm. Such SMs only handle the process variabilities, resulting in accurate models obtained with less simulations.
2) A numerical testing performed on two real consumer electronics circuits provided by STMicroelectronics, a worldwide established company specialized in the design and production of circuit for consumer electronics. Such experimental results also include the comparisons with the results obtained by using WiCkeD, the industrial tool currently employed by STMicroelectronics for circuit design.

The DFO algorithm we adopt in our numerical testing is DFL-box [11], a mixed-integer line-search based optimization method. We are interested in mixed-integer methods because often discrete parameters must be considered in analog sizing (see [12], [13]).

In the following, we will refer to our method for yield optimization as the SVM-DFO method.

II. Problem Formulation

The aim of this paper is to determine the design variables x_d in such a way that in the production process the yield is maximized. The yield corresponding to a design x_d, under given operating conditions $\bar{x}_o$ is formally defined as

$$Y(x_d) = \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} \delta\big(x_p\big) \cdot \mathrm{pdf}\big(x_p\big) \cdot dx_p = E\big\{\delta\big(x_p\big)\big\} \quad (3)$$

where

$$\delta(x_p) = \begin{cases} 1, & x_p \in A_p \\ 0, & \text{otherwise} \end{cases}$$

with

$$\mathcal{A}_p = \big\{x_p \mid l_i \le f\big(x_d, \bar{x}_o, x_p\big) \le u_i, i = 1, \dots, m\big\}$$

and $\mathrm{pdf}(x_p)$ denotes the probability density function of the process variables x_p. The operating variables $\bar{x}_o$ act as an external input, and affect the yield according to their values.

As it is not possible to calculate analytically the integral in (3) we measure the yield by means of the expectation value

$$\hat{Y}(x_d) = \hat{E}\big\{\delta\big(x_p\big)\big\} = \frac{1}{n_s}\sum_{\mu=1}^{n_s} \delta\Big(x_p^{(\mu)}\Big) = \frac{n_{ok}}{n_s} \quad (4)$$

where $x_p^{(\mu)}, \mu = 1, \dots, n_s$ are normally distributed samples of x_p and n_s is the number of samples in the MC analysis. Therefore, the estimator is given by the number n_{ok} of the samples which satisfy the specifications divided by the total number of samples n_s. It could be possible to use $\hat{Y}$ given by (4) as objective function in the optimization problem, but such objective function is not a continuous function and does not capture the geometry of the problem, as it treats in the same way any point out of $\mathcal{A}_p$ no matter how far it is from the feasible region. Therefore, rather than maximizing $\hat{Y}$ in (4) we minimize the following function:

$$\phi(x_d) = \sum_{\mu=1}^{n_s}\sum_{j=1}^{l}\sum_{i=1}^{m}\Big\{\log\Big(\max\Big\{0, l_i - f_i\Big(x_d, \bar{x}_{o,j}, \bar{x}_p^{(\mu)}\Big)\Big\} + \epsilon\Big) + \log\Big(\max\Big\{0, f_i\Big(x_d, \bar{x}_{o,j}, \bar{x}_p^{(\mu)}\Big) - u_i\Big\} + \epsilon\Big)\Big\} \quad (5)$$

where ϵ is a positive parameter close to zero and l is the number of operative cases. Function (5) penalizes how much the performances of a point of the MC analysis are outside their bounds. The logarithm is used to smooth the max function. This is a strategy generally used to handle zero norm problems like in [14]. Therefore, the problem we will solve is given by

$$\begin{aligned} \min_{x_d} \quad & \phi(x_d) \\ \text{s.t.} \quad & x_d \in X_d, \quad x_d^i \in \mathbb{Z}, \ i \in I_z \end{aligned} \quad (6)$$

where $X_d = \{x_d \in \mathbb{R}^n : l_{x_d}^i \le x_d^i \le u_{x_d}^i, \ i = 1, \dots, n\}$ is the feasible set of the design variables and I_z indicates the set of indexes of the design variables that can only assume integer values.

We point out that formulation (6) is the standard formulation for bound constrained mixed-integer optimization problems. Since the derivatives of function ϕ are not available, we need to resort to a derivative free mixed integer optimization algorithm.

III. SVM Surrogate Model-Based Optimization Procedure

In this section, we explain the procedure to generate the MC analysis used to evaluate the objective function ϕ in (5) and we introduce the DFO algorithm used to minimize it.

As said in Section I, the MC analysis of the circuit is performed using an SM given by SVM. The methods for the generation of the SVM and the screening of the process parameters are explained in detail in [10]. Every time the DFO algorithm needs to evaluate the objective function ϕ in a design point x_d for fixed values of the operating variables $\bar{x}_o$, the procedure shown in Fig. 1 is executed as follows.

1) The current values of the design parameters $\bar{x}_d$ are given in input to the MC generation subroutine together with the desired number n_t of samples used for training the SVM.
2) The subroutine generates a uniformly distributed LHS design of experiment for n_t values of the process variables with a standard deviation equal to $\sigma = 5$. We remind the readers that the LHS is created only in respect to the process variables.
3) The n_t realizations of the process variables are given to the simulator that runs n_t simulations for the circuit performances, creating the training set for the SVM.

Authorized licensed use limited to: Universita degli Studi di Roma La Sapienza. Downloaded on July 29,2022 at 12:10:25 UTC from IEEE Xplore. Restrictions apply.

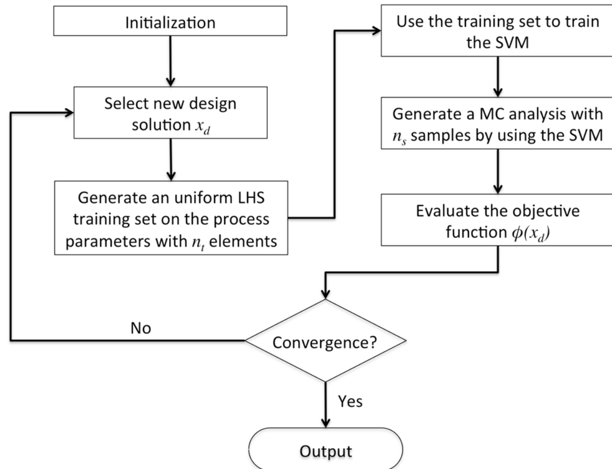

Fig. 1. Yield optimization procedure.

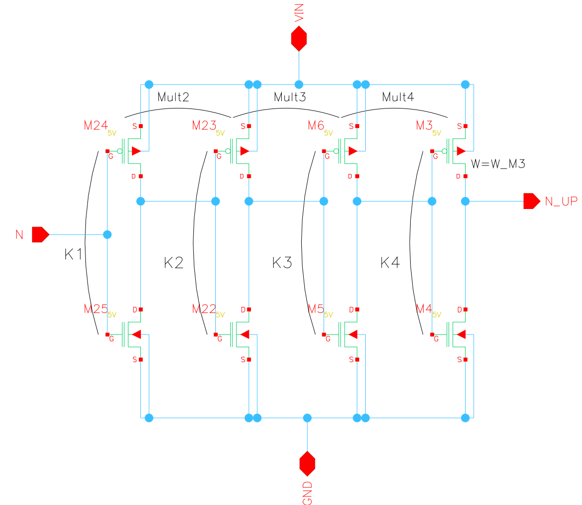

Fig. 2. Circuit variables.

4) The n_t samples are used for training the SVM as SM for the performances of the circuit, given the values of the process variables. For the validation of the model we apply a k-fold cross validation with $k = 5$.
5) The SVM trained model is used instead of the simulator to evaluate an MC analysis of $n_s = 10\,000$ samples of the process variables.
6) The 10 000 values of the circuit performances are used to calculate the value of (5) in $\bar{x}_d$.

As concerns the DFO algorithm for bound constrained mixed-integer minimization of $\phi(x_d)$, we adopt the algorithm described and analyzed in [11]. This algorithm can be thought as a distributed algorithm in the sense that all coordinates are considered cyclically and a different search procedure is adopted depending on the variable type, according to if it is continuous or discrete. Convergence of the algorithm is attained when step lengths $\leq 10^{-3}$ along the coordinate axes corresponding to the design variables do not decrease the objective function ϕ.

It is well known in optimization computations that the most time consuming tasks are the line searches, because in such searches many computations of the objective function are carried out. Therefore, it is of main concern to reduce the number of time consuming simulations required to perform a reliable MC analysis of the circuit performances subject to the process variations. The use of the SVM as SM of the circuit is precisely intended to overcome this main difficulty.

IV. Experimental Results

In this section, we report the numerical results obtained for two circuits developed by STMicroelectronics, a dc–dc converter, and a chain of 15 buffers. The numerical tests are performed at the STMicroelectronics factory in Catania on a computational grid with more than 800 processors. The processors used for the simulations are chosen according to the load on the grid and different processors do not have the same performances. Therefore, we evaluate the performance of the method using the total number of simulations performed, since the main computational load of the procedure is due to the simulations.

In order to ascertain the accuracy of the method we have performed several tests lowering the number of n_t samples in the training set of the SVM. First, we have performed tests with $n_t = 50$ training samples, which we considered, by looking at the number of process parameters in both circuits (9 in the dc–dc converter and 13 in the chain of buffers), a suitable number of samples in order to have an accurate SVM SM. Then, we have performed tests with 40, 30, 20, and 10 samples, tests that we report here only partially.

We first compute the optimal design parameters using the SVM-DFO algorithm. Given these parameters we evaluate the yield in an MC analysis with 10 000 samples using first the SVM SM and then the circuit simulator (CS). Finally, we compare the yield obtained by the SVM-DFO method and the yield obtained by the circuit designers at STMicroelectronics using WiCkeD.

In these experiments, the operating variables $\bar{x}_0$ are fixed to their worst case values, rather than to their nominal values, found by using a worst case operating parameters optimization [2]. This is a conservative practice adopted by STMicroelectronic, based on the assumption that if the performance specifications are satisfied at the worst cases, they are satisfied for any other feasible values of the operating variables.

In the optimization procedures the initial point, provided by the circuit designers, is the same for SVM-DFO and WiCkeD.

A. DC–DC Converter

The dc–dc converter of concern supplies power to different circuits composing AMOLED displays in portable consumer electronics devices. We are interested in the optimal design of a specific section of the converter, an integrated circuit shown in Fig. 2 and given by:

1) a chain of four CMOS inverter;
2) the high side (pMOS) and the low side (nMOS) output stages;
3) the driving signals N_UP, LX1.

As concerns the variables, we have the following variables.

1) *Eight Design Variables:*
   a) *K1–K4:* The scale factor between pMOS and nMOS (discrete);
   b) *Mult2–Mult4:* The scale factor along the inverter chain (discrete);
   c) *W_M3:* The width of the last pMOS inverter in the chain (continuous).

   The first seven design variables (i.e., $K1$–$K4$, Mult2–Mult4) can assume only the integer values $x_d^i \in \{1, 2, \ldots, 10\}$, $i = 1, \ldots, 7$. The last variable, W_M3, must satisfy the bound constraint $10^{-3} \leq W_M3 \leq 1.6 \cdot 10^{-3}$.

   We remark that the width of a specific component in the chain can be easily obtained by the width of the last pMOS (e.g., $W_M4 = W_M3/K4$, $W_M6 = W_M3/\text{Mult4}, \ldots$).

Authorized licensed use limited to: Universita degli Studi di Roma La Sapienza. Downloaded on July 29,2022 at 12:10:25 UTC from IEEE Xplore. Restrictions apply.

TABLE I
RESULTS FOR THE DC–DC CIRCUIT WITH DIFFERENT SIZES OF THE SVM TRAINING SET

n_t	Iter.	Sim.	SM Yield	CS Yield
10	86	4300	0.8531	0.875
30	86	12900	0.8305	0.875
50	86	21500	0.8314	0.875

TABLE II
OPTIMAL DESIGN PARAMETERS FOR THE DC–DC CIRCUIT

	x_d^*
SVM-DFO , $n_t = 10 - 50$	$(1, 3, 2, 2, 9, 3, 6, 1.6 * 10^-3)$
WiCkeD	$(1, 3, 1, 2, 6, 3, 7, 1.6 * 10^-3)$

2) *Two Operating Variables:*
   a) *V, T:* Supply voltage and temperature.
3) *Nine Process Variables:* The process variables related to the pMOS and nMOS devices after a sensitivity analysis have been screened to nine, characterized by a Gaussian distribution with mean value 0 and standard deviation equal to 1.

As concerns the performance features, they are given by the delays of the circuit as follows.

1) *Three Performance Features:*
   a) *Delay$_1$:* It is the delay in turning on the circuit; for this performance lower and upper bounds are $l_1 = 0$ ns, $u_1 = 21$ ns.
   b) *Delay$_2$:* It is the delay in turning off the circuit; lower and upper bounds are the same of Delay$_1$.
   c) *Delay$_S$:* It is the delay symmetry defined as Delay$_1$ − Delay$_2$. It is the performance, the designers are most interested in. Lower and upper bounds are $l_3 = -3.15$ ns, $u_3 = 3.15$ ns.

The values of the operating variables are fixed at four worst cases values as follows.

1) *Worst case for Delay$_1$ and Delay$_2$ at the lower bound of the performances:* $V = 2.3$ V, $T = 120$ °C.
2) *Worst case for Delay$_1$ and Delay$_2$ at the upper bound of the performances:* $V = 4.8$ V, $T = 120$ °C.
3) *Worst case for Delay$_S$ at the lower bound of the performance:* $V = 2.3$ V, $T = -40$ °C.
4) *Worst case for Delay$_S$ at the upper bound of the performance:* $V = 4.8$ V, $T = -40$ °C.

In Table I we report the following.

1) The number of samples used for training the SVM as SM of the circuit.
2) The number of the iterations of the DFO algorithm to reach the optimal design point x_d^*.
3) The number of simulations required in the optimization procedure.
4) The yield obtained by the SM with an MC analysis over 10 000 samples of process parameters.
5) The yield obtained by the CS using the same 10 000 samples.

In Table II, we compare the design point x_d^* and the design point found by WiCkeD. We notice that the SVM-DFO algorithm reaches the same optimal solution no matter the number of samples in the training set. This behavior indicates that the method is not affected by the decreasing accuracy of the SM.

In Table III, we report the results of the SVM SM-based method compared to those obtained using WiCkeD. In particular, we report the yield resulting for each performance and the total yield. It can be seen that no method is superior to the other. WiCkeD performs better for Delay$_1$ upper and slightly better for Delay$_2$ upper, while SVM-DFO fares better for Delay$_2$ lower and for Delay$_S$ upper. The difference in performance between the two methods in the Delay$_1$ upper case makes the total yield of WiCkeD superior. On the other side, the most important performance is the Delay$_S$ making the solution obtained by SVM-DFO better from a design point of view because it reaches the 100% of yield. As concerns the efficiency of SVM-DFO, it reaches the optimal solution in much less simulations than WiCkeD, therefore obtaining a valuable design point with much less computational effort.

TABLE III
COMPARISONS OF THE RESULTS BETWEEN SVM-DFO AND WICKED

Performance	Temp.	Voltage	Y. WiCkeD	Y. SVM-DFO
Delay 1 Lower	120	2.3	100%	100%
Delay 1 Upper	120	4.8	92.54%	88.26%
Delay 2 Lower	120	2.3	100%	100%
Delay 2 Upper	120	4.8	95.80%	93.56%
Delay S Lower	-40	2.3	99.26%	100.00%
Delay S Upper	-40	4.8	96.54%	100.00%
Total Yield			90.12%	87.49%
Total Simulations			11000	[4300, 21500]

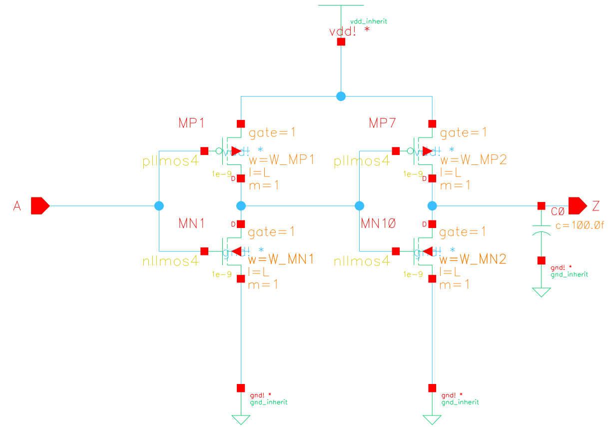

Fig. 3. Cell in the chain of buffers.

B. Chain of Buffers

The second circuit considered is a chain of 15 buffers that are used to generate a programmable delay of the input signal. A signal can be delayed by a programmable quantity by switching on or off each buffer in the chain. The buffers are composed by two inverters and consists of four MOS, two nMOS, and two pMOS devices, as shown in Fig. 3.

The design parameters are the widths W and lengths L of the four MOS devices constituting a buffer in the chain and the goal of the optimization process is to minimize the low-to-high and high-to-low propagation delays as well as their difference, and the power dissipated by the circuit when all the 15 buffers of the chain are on. Once the optimal buffer size has been found, all buffers in the chain will have this optimal size.

The variables of the circuit are as follows.

1) *Five Design Variables (Continuous):*
   a) *WMN$_i$:* Width of transistor MN$_i$, with $10\ \mu\text{m} \leq \text{WMN}_i \leq 300\ \mu\text{m}$, $i = 1, 2$.
   b) *WMP$_i$:* Width of transistor MP$_i$, with $10\ \mu\text{m} \leq \text{WMP}_i \leq 300\ \mu\text{m}$, $i = 1, 2$.
   c) *L:* Length of transistors, with $0.28\ \mu\text{m} \leq L \leq 0.40\ \mu\text{m}$.
2) *Two Operating Variables:* The voltage V and the temperature T.

Authorized licensed use limited to: Universita degli Studi di Roma La Sapienza. Downloaded on July 29,2022 at 12:10:25 UTC from IEEE Xplore. Restrictions apply.

TABLE IV
RESULTS FOR THE CHAIN OF BUFFERS WITH DIFFERENT SIZES OF THE SVM TRAINING SET

n_t	Iter.	Sim.	SM Yield	CS Yield
20	108	2160	0.983	0.963
50	112	5600	0.977	0.953

TABLE V
OPTIMAL DESIGN PARAMETERS FOR THE CHAIN OF BUFFERS

	x^{*1}_d	x^{*2}_d	x^{*3}_d	x^{*4}_d	x^{*5}_d
SVM-DFO					
n_t =20	2.816E-05	4.112E-05	4.068E-05	3.904E-05	4.154E-07
n_t =50	2.812E-05	4.118E-05	4.068E-05	3.906E-05	4.154E-07
WiCkeD	2.820E-05	4.174E-05	4.074E-05	3.918E-05	4.144E-07

TABLE VI
COMPARISONS OF THE RESULTS BETWEEN SVM-DFO WITH $n_t = 20$, $n_t = 50$, AND WICKED

Performance	Y. WiCkeD	Y. SVM-DFO	Y. SVM-DFO
		$n_t = 20$	$n_t = 50$
t_{HL} Lower	100%	100%	100%
t_{HL} Upper	99.2%	99.5%	99.2%
t_{LH} Lower	100%	100%	100%
t_{LH} Upper	98.8%	99.4%	98.8%
t_D Lower	99.6%	98.1%	98.9%
t_D Upper	93.7%	98.4%	97.5%
Power Lower	100%	100%	100%
Power Upper	99.9%	99.9%	99.9%
Total Yield	92.1%	96.3%	95.3%
Simulations	10000	2160	5600

3) *13 Process Variables:* The process parameters have been screened to 13 variables affecting this circuit.

We consider $m = 4$ performance features as follows.

1) t_{HL}: High-to-low propagation delay, with 100 ps $\leq t_{\mathrm{HL}} \leq$ 3 ns.
2) t_{LH}: Low-to-high propagation delay, with 100 ps $\leq t_{\mathrm{LH}} \leq$ 3 ns.
3) t_D: $t_{\mathrm{HL}} - t_{\mathrm{LH}}$, with 0 s $\leq t_D \leq$ 20 ps.
4) pw: The power dissipated by the circuit, with 4 mW $\leq$ pw $\leq$ 6 mW.

From the preprocessing on the circuit, we can observe that the higher the temperature is, the larger the two delays become. Therefore, we only need to analyze the circuit with the temperature at its upper bound $T = 150\ ^\circ C$. The voltage is set at the value $V = 1.2$ V.

The results obtained by using the method are presented in Tables IV and V. In the last row of Table V, we also report the optimal design point found by WiCkeD. We notice that for $n_t \geq 20$ the accuracy of the SVM SMs in predicting the yield obtained by circuit simulations is quite satisfactory, with a value of the yield given by the optimal design point larger than 95%. We notice also that the optimal design parameters found are all close to each other, with changes only at the third significant digit. This shows a satisfactory level of reliability in the proposed methodology: even with really few samples in the training set, and a limited computational effort, it is possible to find good enough design parameters.

The comparison with the results obtained using WiCkeD is reported in Table VI and is quite encouraging. In the second column, we report the results of WiCkeD for every performance and the number of simulations, while in the third and fourth columns we report similar results for SVM-DFO with $n_t = 20$ and $n_t = 50$ in the training set.

It is possible to see that not only SVM-DFO uses significantly less simulations than WiCkeD to find a suitable solution, but also that the yield of the SVM-DFO is rather superior to the one obtained by WiCkeD. In detail, only for the t_D at the lower bound the yield of the SVM-DFO is lower by a significant percentage, while at the upper bound we observe a substantial improvement of SVM-DFO with respect to WiCkeD in all the considered runs.

V. CONCLUSION

We have presented a novel approach for yield optimization in electronic circuit design that combines an accurate SM with an efficient mixed-integer nonlinear derivative-free optimization algorithm. The method has been experimented using two real consumer electronic circuits, and the optimal design variables show high values of the yield. The method also shows a good behavior when only a limited number of samples are used to train the SVM, and compares well with WiCkeD. Taking into account the fact that WiCkeD is being developed as a commercial tool since more than 15 years, we can conclude that the method described in this paper in its first stage of experimentation looks quite promising and worth of further development effort.

REFERENCES

[1] A. Ciccazzo, V. Latorre, G. Liuzzi, S. Lucidi, and F. Rinaldi, "Derivative-free robust optimization for circuit design," *J. Optim. Theory Appl.*, vol. 164, no. 3, pp. 842–861, 2015.

[2] H. E. Graeb, *Analog Design Centering and Sizing*. Dordrecht, The Netherlands: Springer, 2007.

[3] K. J. Antreich, H. E. Graeb, and C. U. Wieser, "Practical methods for worst-case and yield analysis of analog integrated circuits," *Int. J. High Speed Electron. Syst.*, vol. 4, no. 3, pp. 261–282, 1993.

[4] K. J. Antreich, H. E. Graeb, and C. U. Wieser, "Circuit analysis and optimization driven by worst-case distances," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 13, no. 1, pp. 57–71, Jan. 1994.

[5] K. J. Antreich *et al.*, "WiCkeD: Analog circuit synthesis incorporating mismatch," in *Proc. IEEE Conf. Custom Integr. Circuits*, Orlando, FL, USA, 2000, pp. 511–514.

[6] S. Basu, B. Kommineni, and R. Vemuri, "Variation-aware macromodeling and synthesis of analog circuits using spline center and range method and dynamically reduced design space," in *Proc. 22nd Int. Conf. VLSI Design*, New Delhi, India, 2009, pp. 433–438.

[7] M. Stein, "Large sample properties of simulations using latin hypercube sampling," *Technometrics*, vol. 29, no. 2, pp. 143–151, 1987.

[8] A. Singhee, S. Singhal, and R. A. Rutenbar, "Practical, fast Monte Carlo statistical static timing analysis: Why and how," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, San Jose, CA, USA, 2008, pp. 190–195.

[9] B. Liu, F. V. Fernández, and G. G. E. Gielen, "Efficient and accurate statistical analog yield optimization and variation-aware circuit sizing based on computational intelligence techniques," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 6, pp. 793–805, Jun. 2011.

[10] A. Ciccazzo, G. Di Pillo, and V. Latorre, "Support vector machines for surrogate modeling of electronic circuits," *Neural Comput. Appl.*, vol. 24, no. 1, pp. 69–76, 2014.

[11] G. Liuzzi, S. Lucidi, and F. Rinaldi, "Derivative-free methods for bound constrained mixed-integer optimization," *Comput. Optim. Appl.*, vol. 53, no. 2, pp. 505–526, 2012.

[12] M. Pehl and H. Graeb, "RaGAzi: A random and gradient-based approach to analog sizing for mixed discrete and continuous parameters," in *Proc. 12th Int. Symp. Integr. Circuits (ISIC)*, Singapore, 2009, pp. 113–116.

[13] M. Pehl and H. Graeb, "Tolerance design of analog circuits using a branch-and-bound based approach," *J. Circuits Syst. Comput.*, vol. 21, no. 8, pp. 1–17, Dec. 2012.

[14] F. Rinaldi, F. Schoen, and M. Sciandrone, "Concave programming for minimizing the zero-norm over polyhedral sets," *Comput. Optim. Appl.*, vol. 46, no. 3, pp. 467–486, 2010.

Authorized licensed use limited to: Universita degli Studi di Roma La Sapienza. Downloaded on July 29,2022 at 12:10:25 UTC from IEEE Xplore. Restrictions apply.