

A Best Balance Ratio Ordered Feature Selection Methodology for Robust and Fast Statistical Analysis of Memory Designs

Lama Shaer, *Student Member, IEEE*, Rouwaida Kanj^{ib}, *Senior Member, IEEE*,
and Rajiv V. Joshi^{ib}, *Life Fellow, IEEE*

Abstract—Recently, machine learning yield models for integrated circuit (IC) have gained widespread prominence in the EDA community, and are very promising in terms of emulating memory design functionality and thereby speeding up circuit simulation-based variance reduction methods. A main challenge that arises in this area is a class imbalance that occurs naturally due to the high targeted manufacturing yield. Thus, the imbalanced nature of the sampled memory datasets can compromise the model performance. In this work, we attain deep insights into the memory classification problem for modeling rare fail events in the context of importance sampling-based yield analysis. We propose a comprehensive and computationally efficient method that addresses the joint considerations of the best combination of relevant features and class balance ratios, which are key for classifier generalization capability. The methodology relies on synthetic minority over-sampling techniques to enforce the minority class while probing for the best data balance ratio in conjunction with an iterative L_1 -SVM-based approach that qualifies as an approximation to the L_0 -norm regularization for the best feature subset selection. We compare the proposed methodology against standalone L_1 -SVM solutions, unbalanced L_0 -norm approximation as well as an algorithmic data balancing method in the context of yield estimation methodology. The methodology is shown to result in high fidelity classifiers as demonstrated when analyzing the yield of a 14-nm FinFET SRAM cross-section with speedup of 179 \times for the importance sampling simulations compared to pure circuit simulation-based approaches and an average error of 0.19 σ .

Index Terms—Data Balancing, importance sampling, regularization, support vector machine (SVM), synthetic minority over-sampling technique (SMOTE), yield.

I. INTRODUCTION

WHILE aggressive technology scaling has been a major driver behind the semiconductor industry, it has exacerbated the process, voltage, and temperature variations for

nano-scale technologies. The unavoidable uncertainties that arise result in adverse effects on the circuit performance and functionality [1]. Specifically, this is very true for dense memory arrays that utilize the smallest devices on the chip and thus tend to be most sensitive to these variations. With millions of cells in an array, designers enforce tight constraints on cell fail probabilities with requirements of less than one-part-per-million fails. Capturing these fail probabilities with high-fidelity is an indispensable part of the design cycle.

Monte Carlo analysis is traditionally used as a statistical means to estimate design failure probabilities. However, the Monte Carlo simulation method is not practical for estimating such rare fail probabilities as it would require millions of SPICE simulations to capture a single failing point [2]. It is therefore critical to develop fast and reliable statistical methods for the purposes of rare event estimation. Several methodologies have been proposed to speed up memory rare event estimation. These include methods that involve developing statistical models for the SRAM read and write DC noise margin [3]. Mukhopadhyay et al. [4] developed models for the SRAM cell read and write access times distributions. Singhee and Rutenbar [2] proposed Statistical Blockade as a fast statistical methodology that relies on extreme value theory to sample and model the tail distribution probabilities. Circuit-simulation-based importance sampling methods [5], [6], [7], [8] that capture the cell dynamic margins are popular variance reduction techniques that were developed to efficiently accelerate Monte Carlo simulations for rare fail estimation of SRAM cell dynamic margins. By biasing the sampling toward the critical fail region, importance sampling methods offer several orders of magnitude speedup over traditional Monte Carlo [5]. Accurately estimating rare memory fail probabilities is a challenging task that is closely related to machine learning. Recently, machine learning techniques gained major interest in the area of memory design to develop accurate and fast yield models for memory design yield enhancement [9]. Often functional fails translate into discontinuities in the design metric space. In [10], we proposed a regularized logistic regression-based framework with ordered feature selection (OFS) to capture the binary nature of the SRAM fail mechanisms. Indeed, machine learning has become a vital tool for modeling process variation effects and accurate yield estimation. This area, nevertheless, faces several challenges due to the rare fail event nature of the array yield estimation problem [11]. One such challenge is identified as the class imbalance problem. In fact, a class imbalance is a classical problem which is dominant in many real-world scenarios, such as anomaly detection, email fraud detection, and computer intrusion detection. Often, “real” datasets are imbalanced and are dominated by a group of “normal” sample points existing along with

Manuscript received 25 April 2021; revised 7 December 2021 and 14 August 2022; accepted 27 September 2022. Date of publication 12 October 2022; date of current version 19 May 2023. This work was supported by the MSFEA Dean’s Office at the American University of Beirut. This article was recommended by Associate Editor Y. Shi. (Corresponding author: Rouwaida Kanj.)

Lama Shaer was with the Electrical and Computer Engineering Department, American University of Beirut, Beirut 1107 2020, Lebanon. She is now with Hyundai Motors America, Fountain Valley, CA 92728 USA (e-mail: las24@aub.edu.lb).

Rouwaida Kanj was with the Electrical and Computer Engineering Department, American University of Beirut, Beirut 1107 2020, Lebanon. She is now with Synopsys Research and Development, Mountain View, CA 94043 USA (e-mail: rk105@aub.edu.lb).

Rajiv V. Joshi is with IBM TJ Watson, New York, NY 10562 USA (e-mail: rvjoshi@us.ibm.com).

Digital Object Identifier 10.1109/TCAD.2022.3213762

1937-4151 © 2022 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.

See <https://www.ieee.org/publications/rights/index.html> for more information.

a small group of minority “abnormal” sample points [12]. Since most conventional machine learning techniques postulate balanced data distributions, their performances may not be necessarily optimal when the dataset used is imbalanced [13]. Particularly, the aspect of rarity is inherent to the SRAM cell fail mechanisms. This naturally results in data imbalance where the abnormal sample points are fail sample points, and misclassifying them as normal can impact the accuracy of the yield estimate. This is especially true for methodologies, such as [10] and [14] that rely on thresholding or the classifier model outcome to compute the fail probabilities of the importance sample points which hinge on and around the fail boundary. Thus, it is crucial to address the issue of data imbalance [15] and properly model the failure boundary for accurate statistical analysis of memory designs.

Support vector machine (SVM) is a popular supervised machine learning methodology that is well known for its efficiency, prediction accuracy, and robustness [16]. SVMs have been very effective in dealing with balanced datasets, and it has been shown that they do not perform “too badly for moderately skewed data sets” compared to other machine learning algorithms [17]. SVM could lead to “suboptimal results” when dealing with imbalanced datasets [18]. There are many reasons why SVMs are still affected by class imbalance. Particularly, due to the sparseness of the minority class sample, the low presence of minority examples makes them appear further from the ideal class boundary than the majority ones. In an effort to reduce misclassification, and due to class imbalance, the model may result in a separating hyperplane that is highly skewed toward the minority class, and this may in turn lead to low performance on minority sample points. It has been experimentally demonstrated [19] that the ratio of the majority to minority support vectors becomes also imbalanced in an imbalanced dataset. Therefore, a test sample point lying close to the boundary will be surrounded by more majority support vectors. Hence, the model is more likely to classify a boundary point as a majority point particularly for nonseparable boundary cases [19], [20]. Other perspectives, though, have been debated. Akbani et al. [17] stated that the coefficients of the minority support vector may be inherently larger reducing to some extent the effect of the data imbalance on the SVM model. To alleviate the data imbalance problems for classifiers in general and SVMs in particular [18], two forms of solutions were carved: 1) algorithmic methods, also known as implicit methods and 2) data handling methods, also known as explicit methods. At the algorithmic level, solutions include adjusting the misclassification costs using different weights for the different classes so as to reverse the data imbalance, such as [21] and “different error cost method” in [22]. The latter aims to avoid any boundary skewness toward the minority sample points. Alternatively at the algorithmic level, the data imbalance problem can be formulated as an anomaly detection problem. Hence, one-class learning methods [23] can be invoked to train a one-class classifier on the majority data set alone, thereby treating the minority sample points as outliers. Such approaches are especially useful when minority data points are highly under represented, or do not follow a specific structure and appear as noisy instances. In some scenarios, minority data points may be used to build the one-class SVM model. Kowalczyk and Raskutti [24] set the majority class cost to zero in the dual error cost approach to build the latter model. The downside is that one-class SVM discards all information about the other class, and it is thus recommended to use ensemble models that include the one-class on minority and one-class on majority. Alternatively, a two-step

approach can be adopted and the classifier threshold must be tuned using both minority and majority classes [23]. Active learning methods that iteratively identify the closest instances to the separating hyperplane among unused training sample points were further proposed in [25]. Several methods have also been proposed to reduce SVM sensitivity to data imbalance by modifying the underlying kernel functions. These include class boundary alignment methods [26] and margin calibration methods [27]. Finally, fuzzy SVM methods [28] were introduced to address the problem of class imbalance in the presence of true outliers and noise at the algorithmic level. Other implicit methods include tuning the boundary to remove the bias toward majority [29] and a 2-D surface solution that efficiently searches for the optimal solution among a family of regularized solutions with variable weights for the imbalanced dataset cost functions [30]. Data handling methods, on the other hand, focus on balancing the ratio between the minority and majority sample points. Thus, they modify the imbalanced data ratios set by utilizing resampling methods [15]. Recently, there have been several advances in data handling schemes for machine learning applications [12], [20], [31], [32]. These include random oversampling approaches, random undersampling approaches, and synthetic sample point generation such as the synthetic minority over-sampling technique (SMOTE). Batuwita and Palade [33] proposed performing oversampling over the critical subset of features for enhanced accuracy. To solve the imbalance problem in SVMs, Lin et al. [34] relied on generating ensemble models. Each of these models is derived by creating multiple datasets from the original majority data set, with or without replacement, to match the size of the minority set; a verdict among the ensemble models outcomes is computed using methods such as majority voting. Other boosting methods have also been used in an ensemble context for learning with class imbalance [11], [35]. While some practitioners advocate for perfectly balanced class distributions, some still believe that the natural balance should be used. Others [36], [37], [38] propose probing for the best class balance stating that the classifier accuracy hinges on the underlying data. Finally, support cluster machines were proposed [39] as an iterative explicit approach that employs a procedure of clustering and shrinking to separate support vectors from nonsupport vector sample points.

Concurrently researchers continue to advocate and stress for the need for L_p -norm regularization for classifier accuracy particularly in the presence of imbalanced datasets emphasizing that these methods have to be used in conjunction with data balancing approaches [40], [41], [42]. Specifically, Chawla [40] and Forman [43] emphasized that for real-world datasets having the high skew level in the class distribution, with the class of interest being relatively rare, it is necessary to select features that lead to a higher separability between the two classes. Thus, in order to accurately capture the skew in the class distribution feature selection-based methodologies need to be applied. Zheng et al. [44] proposed a feature selection framework that rely on the modified filter method. Weston et al. [41] demonstrated effectiveness of L_1 -norm and/or L_2 -norm regularization for SVM accuracy. L_0 -norm approximations were introduced as efficient feature selection methods for SVMs [45].

In this article, targeting to speed up SRAM yield estimation problem, we propose a fast importance sampling yield analysis methodology for memory yield estimation whose simulation engine employs efficient and accurate learning models in the presence of imbalanced datasets. The methodology targets SVM classification as a popular and efficient machine-learning

technique. It relies on 1) SMOTE to enforce the minority class sample points in its effort to probe for the best data balance ratio (BBR) and 2) OFS as an approximation to the L_0 -norm regularization. The latter is implemented through a bi-level optimization formulation where the outer loop employs OFS wrapped around an inner L_1 -SVM loop for enhanced model capabilities. Specifically, our contributions are as follows.

- 1) At the core of the methodology lies a data balancing scheme with varying balance ratio to handle the class imbalance problem due to the typically low proportion of minority fails in memory design problems. This allows for identifying the Best Balance Ratio among a set of ratios ranging between the natural and fully balanced dataset distribution for enhanced classifier generalization capabilities, thereby not ruling out the naturally occurring class distribution in accordance with literature.
- 2) We revise the OFS approach [10] in the context of SVM. We reformulate it as an approximation to the otherwise nonconvex SVM L_0 -norm regularization that aims to identify the relevant SVM features for the respective imbalanced datasets with the different adjusted balance ratios. For this, we build upon a semi-smooth Newton coordinate descent L_1 -SVM solution engine [46] to iteratively identify the relevant features for a given dataset. While the L_1 -SVM is often performed as a relaxation to the L_0 -norm in terms of augmenting the optimization constraints, we demonstrate enhanced accuracy for our L_0 -norm approximation.
- 3) We study an implicit 2-D surface solution methodology targeting imbalanced datasets classification [30] that employs adjusting the classifier learning algorithms to better recognize the minority class. The methodology identifies the optimal cost function weights for the SVM classes as determined among a derived family of solutions. We rely on this approach as our reference implicit data balancing methodology and critically evaluate its classification and balancing capabilities compared to the proposed explicit approach.
- 4) We study the yield of state-of-the-art industrial 14-nm FinFET SRAM design. We compare our proposed BBR-OFS methodology to the L_1 -SVM approach, OFS on the natural distribution, and the implicit approach. Our results demonstrate that BBR-OFS outperforms other methods in terms of accuracy while maintaining significant improvement in terms of the runtime efficiency compared to pure circuit simulation-based approach. This is manifested both at the model prediction and yield estimation levels.

This article is organized as follows. Section II presents a background review on Monte Carlo, importance sampling, and the basics of SVMs. Section III presents the proposed Best Balance Ratio OFS methodology. Section IV presents the results and analysis of the proposed methodology when applied to the yield estimation of memory designs. Finally, Section V presents the conclusions.

II. BACKGROUND

In this section, we provide a quick overview of Monte Carlo and importance sampling methods. We also present a review of SVM and loss function regularization.

A. Monte Carlo and Importance Sampling

A cell with a probability of failure P_f , is perceived as a Bernoulli trial and the memory array size, N_{cells} represents

the number of trials. The yield of an array, in the absence of any redundancy, can thus be calculated as follows:

$$\text{Yield} = (1 - P_f)^{N_{cells}}. \quad (1)$$

Typically, however the design tolerates a few failing cells due to the presence of redundancy. Hence, given r redundant columns and assuming that the faulty cells occur in different columns, the array yield can be computed as follows [2]:

$$\text{Yield}_{\text{redund}} = \sum_{k=1}^r \binom{N_{\text{cells}}}{C_k} * P_f^k * (1 - P_f)^{N_{\text{cells}} - k}. \quad (2)$$

For a design characterized by the process variation parameters \mathbf{x} and a performance evaluation function $f_i(\mathbf{x})$, the Monte Carlo estimate for P_f in (3) can be approximated by (4) as the expected value of the indicator function (5) as stated by the law of large numbers [47]. N is the random sample size

$$P_f = \int_R \text{pdf}(x) dx = \int I(x) \text{pdf}(x) dx = E(I(x)) \quad (3)$$

$$\hat{P}_f = \frac{1}{N} \sum_{i=1}^N I(x_i) \quad (4)$$

$$I(x) = \begin{cases} 0, & \text{if } f_i(\mathbf{x}) \leq f_i^0 \\ 1, & \text{otherwise.} \end{cases} \quad (5)$$

Importance sampling gained widespread attention as a variance reduction method to improve the accuracy of the Monte Carlo simulation [5]. It distorts the natural distribution $\text{pdf}(x)$, to generate more sample points in the region of interest. The unbiased probability of fail is derived according to (6); $\text{pdf}_{IS}(x)$ represents the importance sampling distribution

$$\int f(x) \text{pdf}(x) dx = \int f(x) \frac{\text{pdf}(x)}{\text{pdf}_{IS}(x)} \text{pdf}_{IS}(x) dx \quad (6)$$

$$\hat{P}_f = \frac{1}{M} \sum_{i=1}^M I(x) \frac{\text{pdf}(x_i)}{\text{pdf}_{IS}(x_i)}. \quad (7)$$

Finding the proper $\text{pdf}_{IS}(x)$ is critical for variance reduction. We rely on the approach in [5] that employs a uniform sampling phase to determine the recommended shift for $\text{pdf}_{IS}(x)$ which is used for generating the importance sample points.

B. Review of SVM

SVM was introduced by Vapnik [48]. SVM is a deterministic supervised machine learning method for binary classification. The applications of SVM include face detection, text and hypertext categorization, and bioinformatics [49]. It carries rigorous theoretical advantages and promising computational performance merits in terms of: the ability to be generalized well, and the ability to efficiently find global and nonlinear solutions by relying on the “kernel trick” [16]. For the purpose of understanding and explaining the SVM classifier, we denote the training dataset consisting of n sample points $\{(\mathbf{x}_i, y_i); i = 1, 2, 3, \dots, n\}$ where $\mathbf{x}_i \in \mathbb{R}^d$ is a d -dimensional input feature vector and $y_i \in \{-1, 1\}$ is the corresponding output. The objective of the SVM is to construct an optimal separating hyperplane in the input space between the two classes such that all vectors on one side of the hyperplane belong to the same class as illustrated in Fig. 1 while maintaining a maximal distance to the closest points or vectors. The solution of the SVM is independent of the number of sample points but dependent on the support vectors which are the input vectors

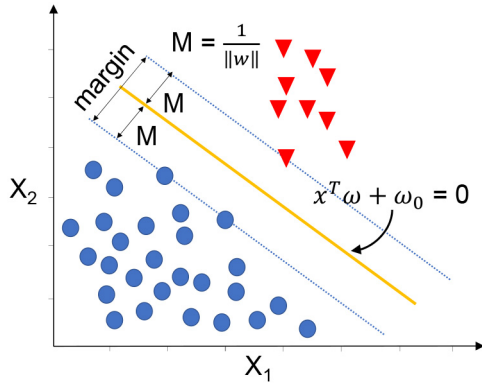


Fig. 1. 2-D illustration of an SVM separating boundary. Support vectors are sample points closest to the boundary.

closest to the hyperplane [50]. If the dataset is linearly separable, the SVM solves for the discriminant function of the separating hyperplane that satisfies $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) > 0$, where \mathbf{w} is a d -dimensional vector and b is a constant representing the hyperplane parameters; $y_i(\mathbf{w} \cdot \mathbf{x}_i + b)$ is the functional margin of the i th training point. A large functional margin reflects good generalizability [51]. The goal is to maximize the minimum margin (8); γ_i is the i th sample point (\mathbf{x}_i, y_i) normalized geometric margin

$$\max \min_i \gamma_i = y_i \left(\left(\frac{\mathbf{w}}{\|\mathbf{w}\|} \right) \cdot \mathbf{x}_i + \frac{b}{\|\mathbf{w}\|} \right). \quad (8)$$

It follows that maximizing the margin is equivalent to minimizing the weights. The linear SVM can thus be reformulated as a convex objective function (9). Note that $(1/\|\mathbf{w}\|)$ represents the distance from the support vectors to the hyperplane

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2} \|\mathbf{w}\|^2 \\ \text{s.t.} \quad & y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 \quad \text{for } i \in 1, 2, \dots, n. \end{aligned} \quad (9)$$

Equation (9) can be solved by using traditional quadratic programming. Often, solutions are derived more efficiently from the dual form employing Lagrange multipliers. When the data is linearly separable, the SVM aims at structural risk minimization and finds the optimal hyperplane that maximizes the sum of the distances to the nearest positive and negative training sample points [52].

When data is not linearly separable, nonlinear SVMs are employed or slack variables are introduced to the optimization to allow some data points to be within the margin. The problem is reformulated as follows:

$$\min_{\mathbf{w}, \xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i \quad (11)$$

$$\text{s.t. } y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i \quad \xi_i \geq 0 \quad \forall i \quad (12)$$

ξ_i is the slack variable that relaxes the margin constraint. This gives way to the 2-norm soft margin SVM classifier where C is a penalty parameter to attain a balance between maintaining $\|\mathbf{w}\|^2$ small and ensuring a good functional margin for most sample points. Hastie et al. [53] adopted an alternative form that sets $\lambda \cong (1/C)$ and (12) can be rewritten as follows:

$$\min_{\mathbf{w}, \xi} \frac{\lambda}{2} \|\mathbf{w}\|^2 + \sum_{i=1}^n \xi_i. \quad (13)$$

λ is a regularization parameter and (13) balances model complexity and goodness of fit. The Lagrange primal becomes

$$L_p : \sum_{i=1}^n \xi_i + \frac{\lambda}{2} \|\mathbf{w}\|^2 + \sum_{i=1}^n \alpha_i (1 - y_i f(\mathbf{x}_i)) - \sum_{i=1}^n \gamma_i \xi_i \quad (14)$$

where $\alpha_i \geq 0$ and $\gamma_i \geq 0$ are the Lagrange multipliers. The system is solved by setting the derivatives to zero along with the Karush–Kuhn–Tucker (KKT) conditions

$$\frac{\delta L_p}{\delta \mathbf{w}} = 0, \quad \frac{\delta L_p}{\delta \xi} = 0 \quad (15)$$

$$\alpha_i (1 - y_i - \xi_i) = 0, \quad \gamma_i \xi_i = 0. \quad (16)$$

Finally, SVMs can also handle nonlinear problems by mapping the original input vectors to a higher dimensional space to solve for the separating hyperplane. This can be performed by explicitly relying on feature mapping functions, $\hat{\mathbf{x}}_i = \phi(\mathbf{x}_i)$, or kernel functions, $K(\mathbf{x}_i, \mathbf{x}_j) \in \mathbb{R}^{d \times d}$, that satisfy the mercer criterion to implicitly perform the mapping [48].

III. PROPOSED BEST BALANCE RATIO ORDERED FEATURE SELECTION METHOD

In this section, we introduce the proposed framework for efficient statistical analysis and modeling for importance sampling-based rare event estimation. Fig. 2 presents the flow diagram of the overall methodology. The methodology relies on importance sampling as its fast Monte Carlo yield estimation engine [5]. To speedup the simulations, at the core of the methodology lies a classification approach that incorporates data balancing along with feature selection for enhanced accuracy in the event of imbalanced datasets. Specifically, the methodology relies on scouting for the best balance ratios in a bi-level optimization approximation to L_0 -norm regularization to enhance prediction capability and reduce simulation cost. For the L_0 -norm approximation, we employ OFS as a wrapper around a fast-newton method-based L_1 -SVM method to guide the model building. We then implement a general framework that identifies the best class balancing ratio in the context of the SVM L_0 -norm approximation.

A. Ordered Feature Selection

Feature selection is critical for “generalization” performance, challenging the curse of dimensionality, and even lowering the time needed to train a model [54]. Particularly, for SVMs, irrelevant features and noise may jeopardize the accuracy, convergence, and efficiency of the classifier [41]. Traditionally, algorithms with feature selection address two rivaling objectives: 1) maximizing the goodness of fit and 2) minimizing the number of variables [54]. Feature selection methods lie in three main categories: 1) the wrapper; 2) the filter; and 3) embedded methods [54]. Methods, such as the wrapper method, however, can be expensive. Alternatively, embedded methods where feature selection is embedded as part of the training algorithm can be used. They are typically more accurate than filter methods and are less prone to overfitting. Traditionally, embedded methods that rely on L_p -norm regularization, $p \in \{1, 2\}$ have been employed [55], where the SVM problem in (13) can be reformulated to a “Loss + Penalty criterion” of the form

$$\min_{f \in H} \sum_{i=1}^n L(y_i, f(\mathbf{x}_i)) + \lambda J(f). \quad (17)$$

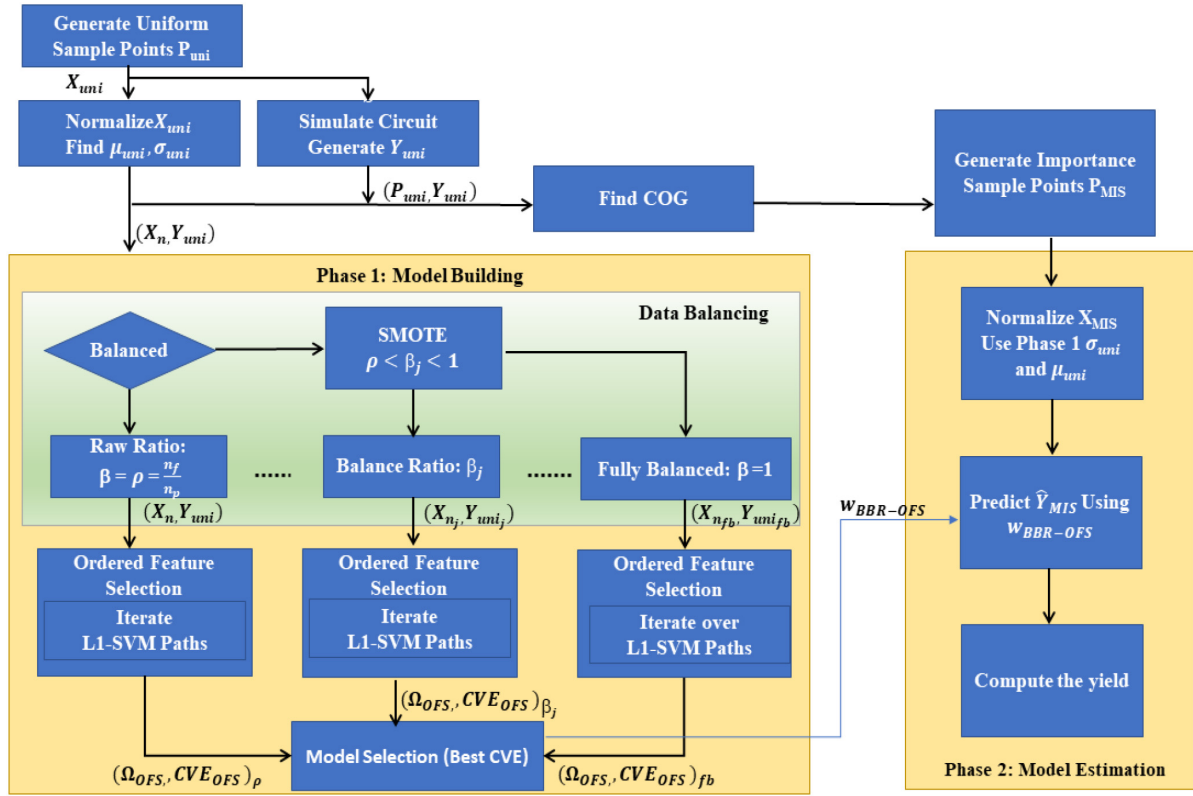


Fig. 2. Methodology flow diagram for the proposed framework.

Algorithm 1 OFS

Input
 X : Input Features
 Y : Output Data

Output
 Ω_{OFS} : OFS Resultant ranked feature subset
 CVE_{OFS} : Minimum cross validation error

- 1: **function** L1-SVM(X, Y)
- 2: Find SVM solution path $\omega(\lambda)$ over $\lambda \in [0, \lambda_{large}]$
- 3: Find $\lambda^{\min} \mid CVE^{\min} = \min(CVE(\lambda^k)) \forall \lambda^k$
- 4: **return** $\omega(\lambda^{\min}), CVE^{\min}$
- 5: **end function**
- 6: **function** OFS(X, Y)
- 7: **Initialization**
- 8: $CVE_{OFS} = \max Num$;
- 9: $\Omega_{OFS} = \{\}$
- 10: Find Solution over all Features X
- 11: $\omega_F(\lambda_F^{\min}) = L1-SVM(X, Y)$
- 12: Sort the $\{\omega_F(\lambda_F^{\min}) \neq 0\}$ model parameters
- 13: $Idx_F, \mid \omega_{sort}(i) = \omega_F(\lambda_F^{\min}, Idx_F(i)) \forall \omega_F \neq 0$
- 14: $j = 1$
- 15: **while** $j < \min(\text{size}(Idx_F), \max NumFeat)$ **do**
- 16: $\Omega_j \leftarrow X(Idx_F(1 : j))$
- 17: $[\lambda_j^{\min}, CVE_j^{\min}] = L1-SVM(\Omega_j, Y)$
- 18: **if** $CVE_j^{\min} < CVE_{OFS}$ **then**
- 19: $CVE_{OFS} = CVE_j^{\min}$
- 20: $\Omega_{OFS} = \Omega_j$
- 21: **end if**
- 22: $j = j + 1$
- 23: **end while**
- 24: **return** Ω_{OFS}, CVE_{OFS}
- 25: **end function**

of f in H_k [56]. Works, such as [57] and [58] cited that L_1 -norm was sufficient to reduce a decent number of variable weights to zero, and least angle regression shrinkage (LARS)- and least absolute shrinkage and selection operator (LASSO)-SVM solvers have been proposed [46], [59]. Weston et al. [41] and Chan et al. [45] proposed L_0 -norm approximations for improved feature selection and enhanced modeling capability. Weston et al. [41] proposed an approximate iterative approach reminiscent of backward elimination. Chan et al. [45] enforced the L_0 -norm approximation by replacing it by a weaker non-convex constraint involving the L_1 -norm and L_2 -norm. Herein, we propose to enforce the best finite set of features Ω and accordingly the L_0 -norm approximation by jointly optimizing the following problem:

$$\min_{\Omega \in F} \min_w \sum_{i=1}^n L(y_i, f(x_i)) + \lambda \|w\|_1 \quad (18)$$

where $L(y, f(x))$ is the huberized hinge loss function described in (19) [60]. $f(x) = wx + b$, and $y \in \{-1, 1\}$ and $h(x) = \text{sign}(f(x))$ is the decision rule used to determine the classification criterion. F is the feature set and Ω represents a feature subset in F . Rather than exhaustively searching for the best subset Ω , we base the outer optimization on an OFS approach similar to [10] as illustrated in Algorithm 1. The methodology initially starts by solving for an L_1 -SVM path, $\omega_F(\lambda)$, over the whole feature set F [61]. Since the regularization parameter λ affects the SVM model parameters tuning over the solution path, we pick the solution $\omega_F(\lambda_F^{\min})$ corresponding to the lowest cross-validation error. As such, we define an ordered index set, Idx_F that represents the features in F sorted in decreasing order based on their respective model parameters, $\omega_F(\lambda_F^{\min})$. This ordered index qualifies as an embedded

$L(y_i, f(x_i))$ is a loss function, and $f(x)$ is an arbitrary function in some Hilbert space H_K , and $J(f)$ is a norm in a Reproducing Kernel Hilbert Space of functions to measure the “roughness”

rank metric. In fact, it has been known that although shrinkage methods, such as LARS and LASSO are not “full-fledged” wrapper methods, these methods are often perceived as semi-wrapper methods with the ability to order features based on their significance [59]. The algorithm then iteratively builds the best feature set, Ω_{OFS} by adding at each iteration the next feature according to Idx_F , and solving for the L_1 -SVM path on the new feature set in a forward selection manner. At each iteration, the best solution along the path is chosen by relying on k -fold cross-validation [62]. The best-ordered feature subset and model for our data is determined as the ordered subset with the lowest cross-validation error (CVE) among all subsets. The OFS flow is then applied to the dataset subject to different balancing ratios to determine the best balance ratio and feature subset combination.

For our purposes, to obtain the L_1 -SVM solution path, we adopt an LASSO-based sparse SVM approach that relies on a semi-smooth newton coordinate algorithm [46], [61] to efficiently solve a huberized hinge loss-based regularization. In fact, it has been shown that reducing the solution complexity relies on identifying the proper loss function [63]. While the hinge Loss function $L(yif(x_i)) = [1 - yif(x_i)]_+$ can be used in (17), it has been shown that the Huberized Hinge loss function in (19), which is almost quadratic, is preferred due to its differentiability. This reduces the computational cost and at the same time maintains linearity similar to the hinge loss for negative margins to prevent outliers from affecting the model

$$L_s^{HH}(yif(x_i)) = \begin{cases} 0, & yif(x_i) > 1 \\ \frac{(1 - yif(x_i))^2}{2\delta}, & 1 - \delta < yif(x_i) \leq 1 \\ 1 - yif(x_i) - \frac{\delta}{2}, & yif(x_i) \leq 1 - \delta. \end{cases} \quad (19)$$

The solution path $w(\lambda)$ is thus piece-wise linear in λ , and the piece-wise linearity “hinges” on the following sample point set partitions associated with L^{HH} [64] that stem from the definition of the loss function (19), i.e., sets 1, 2, and 3 below. The Elbow partition thus represents points that are on the margin (nonbound support vectors), the left partition holds sample points that lie inside the margin (bound support vectors) and the right partition contains the points that lie outside the margin. The solution also hinges on the active feature set $\nu \in \Omega$ that is also function of λ .

- 1) *Right*: $R = \{i \mid yif(x_i) > 1\}$.
- 2) *Elbow*: $E = \{i \mid 1 - \delta \leq yif(x_i) \leq 1\}$.
- 3) *Left*: $L = \{i \mid yif(x_i) \leq 1 - \delta\}$.
- 4) *Active Features Set*: $\nu \mid \{j : w_j \neq 0\}$.

Hence, the solution path according to the KKT conditions starts by adopting a large λ value and proceeds to decrease λ gradually until we reach an event pertaining to a training point hitting E , set of points qualifying as support vectors, or a feature leaving ν , the set of active features with nonzero coefficients. Such events incur a “kink” in the solution and the derivatives of the parameters with respect to λ will change. In between events, the solution for ω is linear in λ .

B. Synthetic Minority Oversampling-Based Data Balancing Approach for BBR-OFS

As discussed earlier, imbalanced classification is one of the most challenging problems encountered in the field of machine learning and has been prevalent in several fields. A dataset is defined as imbalanced when the two classes are not equally represented [65]. Imbalanced datasets have been

shown to compromise the predictive capabilities of the model. For SVM class prediction, it has been demonstrated as discussed earlier that the boundary may be too skewed toward the minority sample points in the event of imbalanced datasets and a point close to minority may be classified as majority. Hence, the weakness of the soft margin can be a cause for the performance loss [17]. In an extreme scenario for imbalanced datasets, smaller $C = (1/\lambda)$ values, i.e., smaller penalty on the slack variable in (13), may result in increasing the margin and causing the SVM to assign all points as majority points. This results in sacrificing the minority points as a consequence of relaxing the margin with the only error to be accounted for being the few minority sample points that are missed. Hence, SVM may in some scenarios not generalize well when addressing imbalanced datasets. To handle the imbalance, researchers have proposed to rely on resampling the datasets (oversampling or undersampling) to reduce the rarity of minority sample points prior to building the classifier. They also recommended to employ in conjunction with data balancing L_p -norm regularization along with appropriate evaluation metrics such as cross validation [66], [67], [68]. Albisua et al. [37] showed that the ideal class distribution is not necessarily the fully balanced distribution and as such attempt to find a class distribution that improves the generalization capability of the classifier. The authors further demonstrate that the results can be enhanced should intelligent resampling techniques be used. In fact, Weiss and Provost [36] showed that the ideal class distribution depends a lot on the learning algorithm used for enhancing the modeling capability. SMOTE is a popular data balancing method shown to improve the model predictive capability in SVMs [65]. In particular, it has been shown to outperform other sampling methods and the quality improvement was measured by different model quality indicators [15]. While simple random undersampling is computationally efficient, it may result in potentially losing information thereby resulting in data becoming noisy and/or biased [11]. In fact, an increase in the undersampling rate can result in an increase in the false positive (FP) rate as demonstrated in [12]. On the other hand, simple oversampling may result in overfitting due to a large number of identical minority sample points [11]. SMOTE overcomes this problem by creating new nonreplicated synthetic minority points thereby enhancing the performance capability of the classifier.

Herein, we employ SMOTE as the resampling technique in our pursuit for the best data balance ratio-based OFS SVM model in Section III-C. We also compare our approach with an algorithmic-based balancing approach for SVMs that is discussed in Section III-D.

1) *Synthetic Minority Over-Sampling Technique*: The synthetic minority over-sampling technique, also known as SMOTE, is a flexible informed data oversampling technique that was introduced in [12] in order to address a class imbalance in binary classification problems consisting of a majority class and a minority class. It aims at increasing the minority sample points by artificially generating synthetic data points in the “feature space” rather than in the “data space” belonging to the minority class. In its effort to properly represent the minority, some assimilated SMOTE to be very much like distortion-based regularization methods [69].

SMOTE generates nonreplicated minority sample points to change the bias of the learner. Based on the desired percentage oversampling, SMOTE selects all or a subset of minority sample points, X_{\min} . For each sample point $X_i \in X_{\min}$, a number of new synthetic minority points is generated based on a predetermined oversampling rate using X_i and one of its k minority

Algorithm 2 Smote Pseudo Code (Adapted From [12])

```

1: procedure SMOTE( $X_{\min}$ ,  $N$ ,  $k$ ) {Returns  $\frac{N}{100} * T$   $x_{syn}$  sample points}
   { $X_{\min}$  is the set of the  $T$  minority sample points to be smoted}
2:   if  $N < 100$  then
3:     Randomize the  $T$  sample points
4:      $T = (N/100) * T$ 
5:      $N = 100$ 
6:   end if
7:    $N = (int)(N/100)$ 
8:    $X_{syn} = \{\}$ 
9:   for  $i = 1$  to  $T$  do
10:    find  $k$  nearest minority neighbor points  $\{X_{nm}\}_i$  for the  $i^{th}$  sample
    point
11:     $X_{syn} \leftarrow X_{syn} \cup \text{GenerateSamplePoints}(N, i, \{X_{nm}\}_i)$ 
12:   end for
13: end procedure

14: function GENERATESAMPLEPOINTS( $N$ ,  $i$ ,  $\{X_{nm}\}_i$ )
15:   while  $N \neq 0$  do
16:    Randomly choose one of the  $k$  nearest neighbor points  $X_j \in$ 
     $\{X_{nm}\}_i$ ,
17:    count++;
18:    for  $l = 1$  to  $numFeatures$  do
19:      Randomly choose  $\delta \in (0, 1)$ 
20:      Generate  $P_{syn}[count][l] = X_j[l] + \delta * (X_j[l] - X_i[l])$ 
21:    end for
22:    Set  $N = N - 1$ 
23:   end while
24:   return  $P_{syn}$ 
25: end function

```

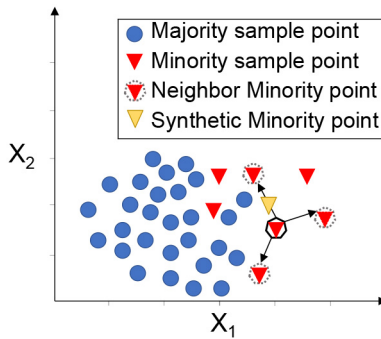


Fig. 3. Synthetic sample point generation (golden triangle) or a specific minority sample point (bold line) using SMOTE.

nearest neighbors, X_{nm_i} (see Fig. 3). To generate the synthetic sample point, the differences between the specific attributes of the sample point under consideration and those of one of its randomly chosen nearest neighbor, $X_j \in X_{nm_i}$, are computed. The differences are then multiplied by random numbers $\delta \in [0, 1]$ and added to the reference sample point attributes. Hence, for each attribute, the following holds:

$$P_{syn}[\text{attr}] = X_j[\text{attr}] + \delta * (X_j[\text{attr}] - X_i[\text{attr}]). \quad (20)$$

The random point attributes are allocated along line segments between the sample point and its nearest neighbor. If the oversampling ratio is larger than 100%, multiple neighbors will be used to generate synthetic points from a given minority sample point according to the SMOTE Algorithm 2 [12]. The runtime complexity for SMOTE is $O(n_f^2)$; for each minority point, the cost to find the k -nearest neighbors is $O(n_f \log_2 k)$, and the time to determine the synthetic sample point is $O(1)$ [70]. Finally, it is worth noting that safe SMOTE methods were introduced in [71] to account for neighboring points from other classes and minimize class overlap.

C. BBR-OFS Framework for Yield Estimation Methodology

The dual optimization in (18) along with the search for the best data balancing ratio comprises the core circuit modeling engine for the proposed yield estimation framework of Fig. 2. Together, they approximate a multilevel optimization that can best be described in

$$\min_{\beta_j} \min_{\Omega \in F} \min_w \sum_{i=1}^n L(y_i, f(x_i)) + \lambda \|w\|_1 \quad (21)$$

where β_j represents a varying data balancing ratio as presented in pseudo code of Algorithm 3 that summarizes the flow of the proposed modeling and yield estimation framework. Thus, the β_j ranges are determined from Phase 1 of importance sampling as follows. The importance sampling yield estimation methodology encompasses two phases: a first uniform sampling phase to determine the center of gravity of fails which comprises the shift center for the distribution in the second importance sampling phase. Similar to [10], our objective is to leverage the circuit simulations performed in the first phase to build a model that can be employed to eliminate the need for circuit simulations in the following phase. For Phase 1, referred to hereon as the model building phase, the original data is preprocessed and the dataset is analyzed to determine the level of imbalance $\rho = (n_f/n_p)$, where n_f is the number of failing (minority) sample points and n_p is the number of passing (majority) sample points. An assorted set of minority oversampling ratios, $\rho \leq \beta_j \leq 1$, spanning the range between the natural raw ratio ρ and the fully balanced one is determined. We rely on SMOTE to generate synthetic sample points corresponding to these ratios. For each resulting dataset with a specific balance ratio, OFS is employed as discussed in Section III-A resulting in multiple models for the different balance ratios. Among these models, the model corresponding to the lowest CVE is adopted. This model thus comprises the BBR-OFS model that employs the critical set of features and minority oversampling ratio that enhance the model and as such enable efficient and accurate yield estimation. In Phase 2, importance sample points generated using the center of gravity of fails as determined in Phase 1 are evaluated and their respective performance metric is derived using the BBR-OFS model. The yield estimate is then derived (7). Maintaining high model fidelity is important especially when using the framework as a comparator among several high-sigma yield design points which can be otherwise a challenging requirement.

D. Note on Implicit 2-D Solution Surface Algorithm for Weighted SVM (2DWSVM)

In classical SVM the data points are treated equally, however in certain scenarios this may not be optimal and applying certain weights to the data points can improve the accuracy of the model. Lin et al. [21] introduced the concept of weighted SVM (WSVM) where the cost function employs a weight parameter π_i to assign different weights to the different classes. The formulation in (17) can be rewritten as (22), and the Lagrange primal for the hinge loss becomes (23)

$$\min_{f \in H} \sum_{i=1}^n \pi_i L(y_i, f(\mathbf{x}_i)) + \lambda J(f) \quad (22)$$

$$L_p = \sum_{i=1}^n \pi_i \xi_i + \frac{\lambda}{2} \|\omega\| + \sum_{i=1}^n \alpha_i (1 - y_i f(\mathbf{x}_i)) - \sum_{i=1}^n \gamma_i \xi_i \quad (23)$$

Algorithm 3 Yield Estimation Methodology

Input
P_{uni} : Process variation parameters for uniform sampling
F_{uni} : Performance metric vector
Output
σ_{IS} : Importance sampling yield estimate based on evaluating response for P_{MIS}
Phase 1 – BBR-OFS Model Building
1: Preprocess Data: Generate polynomial and interaction terms using Hermite functions g_i [75]
2: $X_n = \frac{X_{uni} - \mu_{X_{uni}}}{\sigma_{X_{uni}}}$
3: $Y_{uni} = (F_{uni} > f_0) f_0$ represents fail criterion
4: $X_{n_{min}} = \{X_n (Y_{uni} == 1)\}$ minority set
5: Let $\rho = \frac{n_f}{n_p} = \frac{size(X_{n_{min}})}{size(X_n) - size(X_{n_{min}})}$ raw ratio
6: $\beta \in [\rho : 0.1 : 1]$ $\beta = 1$ fully balanced ratio
7: for $j = 1 : size(\beta)$ do
8: $X_{n_j} = \{X_n \cup SMOTE(X_{n_{min}}, \frac{\beta_j}{\rho} - 1, k = 5)\}$
9: $Y_{uni_j} = \{Y_{uni} \cup \text{Ones}((\frac{\beta_j}{\rho} - 1) * n_f)\}$
10: $[\Omega_{OFS}, CVE_{OFS}]_{\beta_j} = \text{OFS}(X_{n_j}, Y_{uni_j})$
11: end for
12: $\beta_{bbr} CVE_{OFS}_{\beta_{bbr}} = \min(CVE_{OFS}_{\beta_j})$
13: $\Omega_{BBR-OFS} = \Omega_{OFS}_{\beta_{bbr}}$
14: $\omega_{BBR-OFS} = \begin{cases} \omega_{OFS}_{\beta_{bbr}}, & \forall X_n \in \Omega_{BBR-OFS} \\ 0, & \forall X_n \notin \Omega_{BBR-OFS} \end{cases}$
Phase 2 - Model Prediction and Yield Estimation
15: $\mu_{COG} = \text{mean}(\frac{P_{uni} * Y_{uni}}{n_f})$ //Center of Gravity of Fails
16: $P_{MIS} \leftarrow$ Generate process variation importance sample points around μ_{COG}
17: $X_{MIS} = \{g_i(P_{MIS})\}$
18: $X_{n_{MIS}} = \frac{X_{MIS} - \mu_{X_{uni}}}{\sigma_{X_{uni}}}$
19: $Y_{MIS} \leftarrow \text{EvalSVM}(X_{n_{MIS}}, \omega_{BBR-OFS})$
20: Calculate σ_{IS} Yield according to (7)

where weight $\pi_i = \pi$ for the majority class points ($y_i = -1$) and $\pi_i = 1 - \pi$ for the minority class points ($y_i = 1$). $0 \leq \alpha_i \leq \pi$ and $\gamma_i \geq 0$ are the Lagrange multipliers. Similar to Section III-A, the algorithm hinges on the elbow partition, the left partition, and the right partition. $\alpha = 0$ implies a point in the right disjoint set, and $\alpha = \pi$ implies a point in the left disjoint set. For a fixed π value, the standard λ -path algorithm [56] can be employed. A π -path algorithm was also developed to efficiently determine π solutions for fixed λ values [73].

Shin et al. [30] demonstrated that the solution surface is jointly piecewise linear in (λ, π) pair so long that no event happens. An event pertains to a nonsupport vector becoming a support vector or vice versa. Thus, they propose a 2-D solution surface that efficiently computes the entire solution surface in the 2-D (λ, π) space. As λ and π change the sets change and the solutions are continuous. The methodology relies on recursively identifying subregions or polygons, Surf^l defined by their respective vertices $v_r^l = (\lambda_r, \pi_r)$, $r = 1, 2, \dots, n_v$ where n_v signifies the number of vertices as determined by the constraints that prevent an event from happening. The method [30] then exploits the piecewise linearity of the solution to extend the solution until the whole solution surface space is covered. The solution is finer in regions where the solution is complex and coarser otherwise. In addition to the computational efficiency, this offers an advantage over a uniform grid-based solution surface. The methodology can be best summarized as follows. First, we introduce some basic definitions.

- 1) A point $pm^l = (\lambda^l, \pi^l)$ is associated with the corresponding model parameters and partitions $\{\theta^l, E^l, L^l, \text{ and } R^l\}$.
- 2) The maximal polygon Surf^l around pm^l is the set of points $(\lambda, \pi) \in \text{Surf}^l$ such that no event occurs. Hence, within the polygon, the partitions $E = E^l$, $L = L^l$, and $R = R^l$ remain the same and θ can be derived as a linear function of $(\theta^l, \lambda - \lambda^l, \pi - \pi^l)$.
- 3) $\{C\}$ represents the set of linear constraints in (λ, π) space, defined for the different sample points to determine Surf^l around pm^l such that no event occurs; an example of an event, as discussed earlier, is for a sample point $i \in E^l$ to violate the constraint $\theta_i < \pi_i$, or for $i \in L^l$ to violate the constraint $y_i * f(x_i) < 1$.
- 4) Vertices $v_r^l = (\lambda_r^l, \pi_r^l)$ of Surf^l , $r \in [1, n_v]$, are determined by finding the intersection points of the constraints that satisfy all the constraints. These can be identified efficiently due to dominance of some constraints.

The algorithm starts with an initial point $pm^0 = (\lambda^0, \pi^0)$; π^0 is derived from the ratio of minorities to the total number of sample points; λ^0 is calculated from π^0 and the kernel function (linear in our case) as is the case for θ^0 and E^0 [30]. R^0 is the empty set and L^0 is the set of sample points not in E^0 . Based on this initial point, the algorithm identifies an initial Surf^0 and vertices $\{v_r^0\} = \{(\lambda_r^0, \pi_r^0)\}$ along with the respective model parameters for each vertex via interpolation. To build the surface grid, the algorithm iterates as follows. For each pair of adjacent vertices, it finds midpoints pm^l and constructs new surfaces around these midpoints. First, it determines for each midpoint its respective θ values through interpolation based on its initial surface and updates its respective partitions based on the edge which the midpoint violates. It then relies on the resultant model parameters and updated partitions to construct a new set of constraints and determine the respective new polygon and vertices. This continues until the complete surface is covered (boundaries arrived at). Hence, the steps can be defined as follows.

- 1) Given: Input^(k) = $\{pm_{1,1}^{l(k)}, \dots; pm_{1,r_1}^{l(k)}, \dots; pm_{j_k,1}^{l(k)}, \dots; pm_{j_k,r_{j_k}}^{l(k)}\}$ represents the set of midpoints obtained at iteration k and their respective solution sets $\{\lambda^l, \pi^l, \theta^l, E^l, L^l, R^l\}^{(k)}$.
- 2) For the j^{th} midpoint $pm^l \in \text{Input}^{(k)}$.
 - a) Define Surf^l around the midpoint by finding the respective vertices v_r^l , $r \in [1, n_v]$ based on the respective midpoint constraints.
 - b) Update θ_r^l solutions at the different vertices, and save $\text{out}_j = \{\{v_r^l, \theta_r^l\}\}_{r=1 : n_v}$.
 - c) Find the new midpoints for Surf^l , update the partitions at each midpoint, and interpolate the solution for the parameters. Accordingly set a new input $\text{in}_j = \{pm^{l+1}\}_{r=1 : n_v}$ with their respective $\{\lambda^{l+1}, \pi^{l+1}, \theta^{l+1}, E^{l+1}, L^{l+1}, R^{l+1}\}_{r=1 : n_v}$.
- 3) $\text{output}^{(k)} = \bigcup_j \text{out}_j$.
- 4) $\text{input}^{(k+1)} = \bigcup_j \text{in}_j$; remove revisited midpoints.
- 5) Go to step 1 unless input is empty (surface covered).

IV. RESULTS AND ANALYSIS

A. Experimental Setup

We evaluate the performance and assess the merits of the proposed BBR-OFS methodology in the context of state-of-the-art 14-nm FinFET SRAM design with write assist circuitry [74]. Fig. 4(b) presents an illustration of writing a “0”

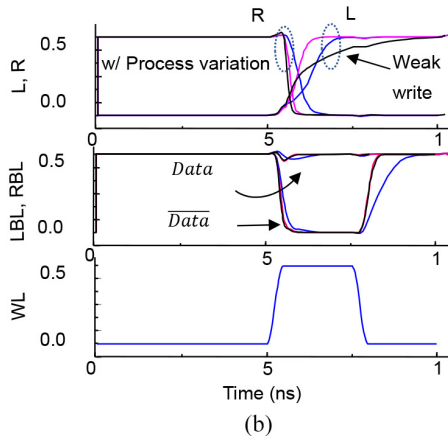
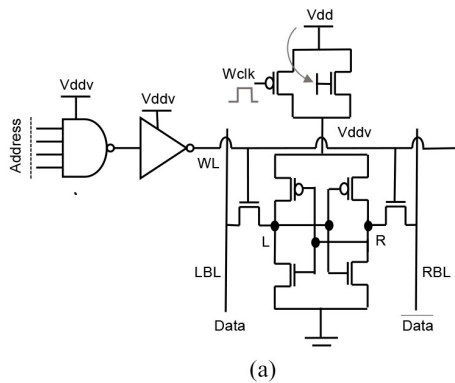


Fig. 4. (a) Schematic of an SRAM cross-section with wordline driver and supply boost. (b) SRAM write operation.

to the right node of the SRAM cell. It is possible, due to process variations, that the cell pullup device becomes strong, and the pass gate becomes weak, thereby leading to a slower write mechanism and in some rare scenarios a writeability fail. We thus define writeability to be the ability to flip the contents of the SRAM cell. In FinFET SRAM designs, writeability suffers in minimum-sized cells due to the quantization effect. Hence, boost and write assist circuitry similar to the one illustrated in Fig. 4(a) are employed to enhance the yield. They enable selectively boosting the supply voltage for the SRAM cell and part of the peripheral logic during low voltage operation. This enhances the yield while maintaining low power operation. Negative bitline boosting can further assist the cell write mechanism by increasing the voltage swing between the true and complement bitlines (LBL and RBL) [74]. For purposes of our analysis, V_{dd} is swept over the range [0.35–0.45 V] for the design with and without boosted write assist for a total of ten experiments whose yield ranges from 2–7 σ . We apply variability to SRAM cell transistors and the critical path devices resulting in a 45-feature vector after generating interaction and second-order polynomial terms [72]. The circuit simulation-based analysis for the different experiments corroborates well with the hardware demonstrating the advantages of the boost.

B. Performance Evaluation: Robustness and Compactness

We compare the performance of the proposed BBR-OFS methodology to the traditional L_1 -regularized SVM solutions, OFS using the unbalanced datasets, as well as implicitly balanced 2-D solution surface WSVM methodology. Our golden reference is the pure circuit simulation-based approach. For

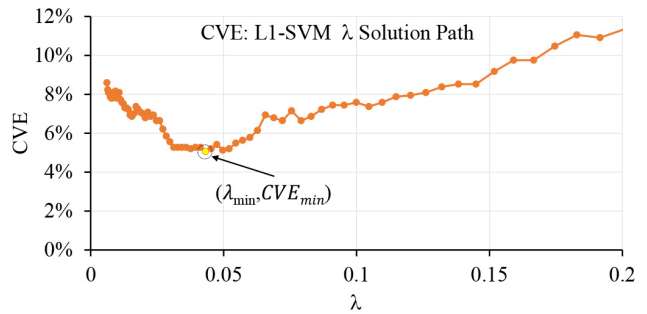


Fig. 5. Cross validation error versus λ for one L_1 -SVM path.

all our experiments, different data balancing ratios were employed, and the BBR-OFS model is used to estimate the yield. The first set of results aims at evaluating the performance of the model. The second set of results aims at evaluating the overall performance of the proposed methodology in the context of yield analysis. Finally, we report on the run-time savings for the proposed methodology. Hereon, we adopt the following terminology.

- 1) L_1 -SVM $_{\rho}$: Refers to the best solution along the LASSO regularized SVM solution path [46]. The subscript refers to employing the natural distribution ratio ρ , of the minority to majority sample points for a given dataset. The original set of features is employed, and sparsity is solely enforced via LASSO regularization.
- 2) OFS $_{\rho}$: Employs OFS that iteratively calls L_1 -SVM solutions to identify the best feature set on the natural datasets; i.e., no data balancing is employed.
- 3) BBR-OFS: Refers to the proposed best balance ratio OFS methodology. OFS is employed with different data balancing ratios, $\rho \leq \beta \leq 1$. Data balancing is performed using SMOTE.
- 4) 2DWSVM: The algorithmic approach proposed in [75] is employed to manage data balancing implicitly through a 2-D (λ, π) solution surface, where π is a weight that emphasizes the minority points in the cost function.

Due to the nature of the importance sampling methodology, each experiment is associated with preprocessed datasets corresponding to the two phases of importance sampling: 1) a uniform sampling dataset P_{uni} and 2) an importance sampling data P_{MIS} set. The model building phase employs the uniform dataset, and the yield estimation Phase 2 employs the importance sampling dataset. As such, we rely on the uniform dataset for model building. To maintain a moderate overhead for the Lp-norm regularizations, we employ 5-fold cross-validation to compute the CVE. We rely on CVE as a holistic metric to provide an unbiased evaluation of different classifier models and balancing ratios during this phase. We evaluate the performance of the model in terms of the yield estimation using the importance sampling dataset which counts as the test data for our purposes. Fig. 5 presents the CVE error as function of λ for an example L_1 -SVM solution path demonstrating the pair $(\lambda_{min}, CVE_{min})$ for one of the experiment. For purposes of our analysis, BBR-OFS relies on the L_1 -SVM path in the search for best-ordered feature sets and balance ratios β_j .

Fig. 6 presents example CVE_{min} values for the fully balanced and raw uniform sampling datasets for the different experiments when their respective ordered features sets are employed near the optimal $\Omega_{BBR-OFS}$ solutions. The impact of SMOTE and feature selection on CVE_{min} is evident. However,

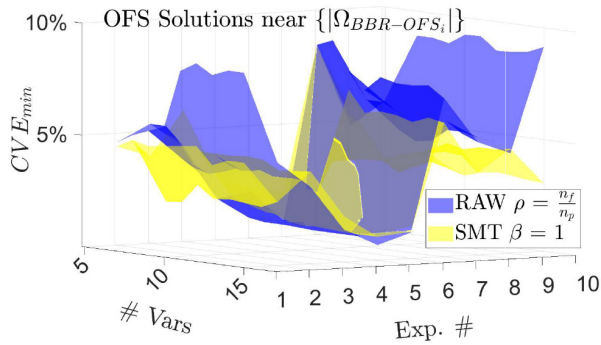


Fig. 6. CVE_{\min} for the different experiments for the fully balanced and raw ratios for the uniform sampling datasets.

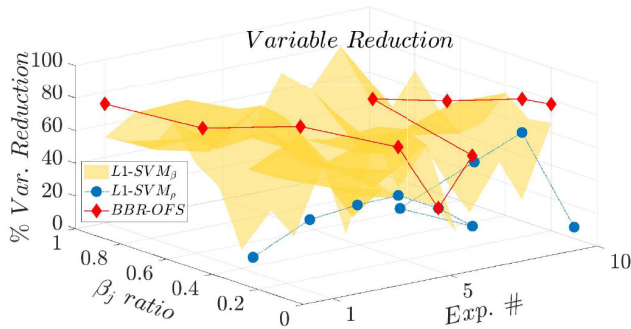


Fig. 7. Variable reduction for the proposed BBR-OFS models is higher than that of $L1-SVM_\rho$, i.e., $L1-SVM$ on raw data and $L1-SVM_\beta$, i.e., $L1-SVM$ on the different balanced datasets.

while in some scenarios SMOTE helped improve CVE_{\min} with the lowest CVE dropping close to 2%, in other scenarios, the raw dataset performed better. Indeed our results, as will be demonstrated later, show that the proper balancing ratio combined with OFS help improve the accuracy thereby enhancing model robustness and improving the prediction at the model building and yield estimation phases.

Fig. 7 presents the percent variable reduction for the proposed BBR-OFS and $L1-SVM_\rho$. It also presents the variable reduction for $L1-SVM_{\beta_j}$, where $L1-SVM$ is applied without OFS to different balancing ratios, $\beta_j^i \in (\rho^i, 1]$; $i \in [1, 10]$ is the experiment number. We note 32.5% average variable reduction for $L1-SVM_\rho$, 46.7% for $L1-SVM_{\beta_j}$ solutions, and 71.5% reduction for the proposed BBR-OFS. The proper balancing ratio together with OFS help remove irrelevant features and enable a sparse solution to minimize overfitting.

C. Performance Evaluation: BBR-OFS AUC and ROC

To demonstrate the efficacy of our BBR-OFS solutions, we evaluate their respective receiver operating characteristic curve (ROC) curves. In addition to CVE, the area under the ROC curve (AUC) has also been used as a metric to assess the classifier performance for imbalanced datasets [76]. It is established that AUC does not favor one class over the other; it is not biased to the majority class and can be used to compare the classification performance for imbalanced datasets.

For a classifier, there are four possible scenarios: 1) a true positive (TP) representing a correctly predicted minority point; 2) an FP for an incorrectly predicted minority point; 3) a true negative (TN) for a correctly predicted majority; and 4) a

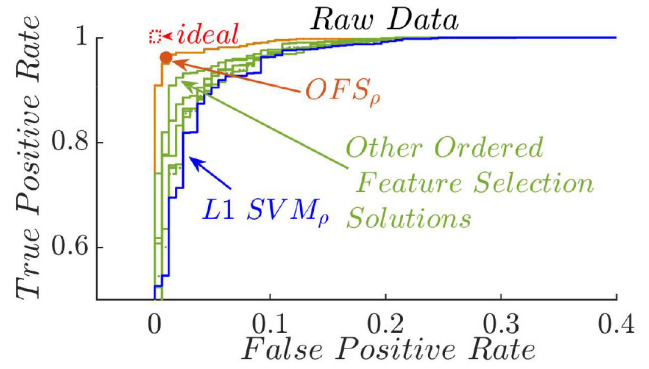


Fig. 8. ROC curves (for $TPR \in [0.5-1]$ and $FPR \in [0-0.4]$) for an example raw dataset indicate improved AUC value for the OFS_ρ solution compared to $L1-SVM$. AUC values for the OFS iterations are also presented.

false negative (FN) for an incorrectly predicted majority. The ROC curve depicts the tradeoff between the TP rate (TPR), or sensitivity, and the FP rate (FPR), or specificity [77]

$$TPR = \frac{TP}{TP + FN}; \quad FPR = \frac{FP}{FP + TN}. \quad (24)$$

AUC can be viewed as the probability of correctly ranking the class “minority”–“majority” sample point pairs. It averages the score of a minority sample point having a higher probability than a majority sample point for all between-class pairs. It is viewed as a rank statistic that can be computed according to the Mann–Whitney–Wilcoxon test (MWW) in (25) without the need for building the ROC curve [78]

$$AUC = \frac{1}{n_f n_p} \sum_{i=1}^{n_p} \sum_{j=1}^{n_f} I(f(x_i^+) > f(x_j^-)) \quad (25)$$

where $I(f(x_i^+) > f(x_j^-))$ is the indicator function that evaluates to one when the embedded inequality holds. For our purposes, TPR corresponds to detecting fail sample points accurately, and a high TPR is critical for the yield estimate not being too optimistic. A high FPR is not desired as it results in a pessimistic yield estimate, especially that FP’s can be close to the fail boundary. Thus, the higher the AUC the better are the results, and the ideal AUC value is 100%. A 50% AUC value indicates that the classifier has bad predictability that is equivalent to that of tossing a coin. For purposes of our analysis, we rely on the ranking of the observations based on the scores (not only the pass/fail labels) to calculate the AUC according to MWW. Thus, we rank each event, i.e., fail point, with respect to other nonevents, passing points, to compute the AUC according to (25) and construct the ROC curves accordingly. Fig. 8 presents the ROC curves for the OFS applied to the raw data for one experiment. It is clear that the AUC for the OFS_ρ solution is better than that for the $L1-SVM_\rho$ solution. Its AUC values are 99.8% for the OFS_ρ and 97.9% for $L1-SVM_\rho$. The figure also depicts the ROCs for several solutions obtained from the iterations of the OFS. It is clear, that with the same underlying dataset, OFS_ρ achieved better performance than the standard $L1-SVM$ method.

Fig. 9 presents the AUC values for the BBR-OFS solution when OFS is applied to the different balancing ratios. The figure thus presents the AUC values for the OFS iterations for two groups which overlap in some cases: 1) best solution group: OFS solutions where each dataset is balanced by its respective best balance ratio as obtained from the BBR-OFS

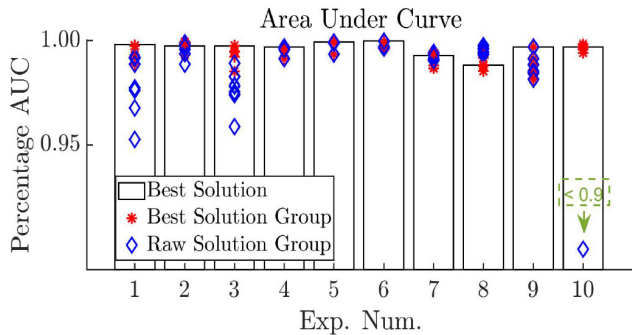


Fig. 9. AUC for the BBR-OFS solution is high compared to the raw dataset solutions and the other solutions employing the same data balancing ratio with varying feature sets. AUC values obtained using MWW [78].

solution and 2) the raw dataset OFS solutions including $L1-SVM_{\rho}$. We report for the BBR-OFS solutions an average AUC value of 99.6% (using MWW), an average TPR of 98.1%, and FPR of 5.3% (latter two relying on SVM margin), compared to Best Solution Group overall average of AUC of around 99.4%, TPR of 97.8%, and FPR of 9.7%. For the raw data group, AUC averaged around 94.2%, and the TPR and FPR averaged 86.7% and 14.6%, respectively. Note for our BBR-OFS solutions, only one experiment favored a fully balanced solution, five favored an intermediate balancing ratio and four sets favored the raw distribution OFS solution. For the raw datasets, we observe AUC values fluctuate in some cases lower than 70% for the $L1-SVM_{\rho}$ solution (not shown here). Clearly, OFS along with optimal balancing ratio help shrink this uncertainty and attain better predictive accuracy-based. BBR-OFS solutions often ranked highest among its group with the balanced datasets demonstrating tighter AUC ranges. As discussed earlier, we view AUC as the probability of the ability of a model to correctly rank the classes. Thus, the figure clearly illustrates the advantage of BBR-OFS in enhancing the AUC values and hence the generalization capability of the classifier.

D. Implicit Versus Explicit Data Balance Ratios: “2-D WSVM”

For the SMOTE data balancing, we varied $\beta_j \in [\rho, 1]$. The 2DWSVM approach explores the (λ, π) 2-D solution surface to address the data imbalance and enable efficient exploration of the piecewise linear parameter solutions in the (λ, π) space. We ran the same datasets through the 2DWSVM approach and identified the best solution pair. Fig. 10 presents the raw ratio ρ compared to the best solution π -adjusted ratio as in (26) where the majority are weighted by π and the minority by $(1 - \pi)$. We note that the best solution-adjusted ratio in the solution space was always close to one, and the respective π value is close to the raw distribution. The best solution CVE ranged between 6% and 25% with an average of 18% for the different experiments, and the CVE error was highest for the datasets with very low (n_f/n_p) ratios

$$\left(\frac{n_f}{n_p}\right)_{\pi_{\text{adj}}} = \frac{(1 - \pi) * n_f}{\pi * n_p}. \quad (26)$$

E. Yield Estimation and Runtime

The methodology relies on the importance sampling-based approach in [5] to estimate the yield. Hence, we utilize

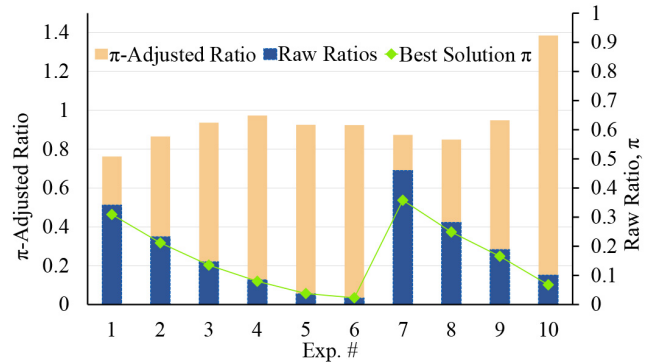


Fig. 10. 2DWSVM π -Adjusted ratio for the best solution.

TABLE I
PHASE II SPEEDUP AND RUNTIME IN SECONDS

Phase	Golden [5]	2D WSVM	L1-SVM	OFS	BBR-OFS
Uniform Sampling	18000	18000	18000	18000	18000
Model Fitting	N/A	Several hrs	0.65	19.39	150.43
Importance Sampling	27000	0.22	0.45	0.4	0.32
Phase II Speedup	1	0.5x	24421x	1364x	179x

the importance sample point response estimates along with importance sampling distribution characteristics to compute the unbiased probability estimates. We compare the results to those obtained using the golden classical circuit simulation-based importance sampling approach in [5]. Hence, for each dataset, we computed the rare event failure probability P_f for the specific failure criteria using Phase 2 performance metric values. We represented P_f by its equivalent σ value

$$\sigma = \phi^{-1}(1 - P_f) \quad (27)$$

where ϕ is the standard normal cumulative density function. Herein, we compare the accuracy and the efficiency of the proposed BBR-OFS methodology for rare fail event estimation. Our reference is the pure circuit simulation-based approach. The BBR-OFS-based yield estimate outperformed the other model-based estimations as emphasized in Figs. 11 and 12 below. Fig. 11 presents the absolute value of yield error $|\sigma_{\text{err}}|$ which is the difference of the yield sigma value between that obtained from the pure circuit simulation-based approach σ_{ref} , and that obtained by relying on the different models for predicting the circuit performance σ_{hat} . The implicit 2DWSVM demonstrated the highest error as was expected based on what was reported earlier in terms of CVE values. OFS_{ρ} outperformed $L1-SVM_{\rho}$ for the yield estimate inline to what has been demonstrated in terms of the AUC-ROC analysis for the model prediction capability. Finally, BBR-OFS also demonstrated both improved average and worst-case error yield sigma errors compared to OFS_{ρ} . Fig. 12 presents the yield estimates for the BBR-OFS, OFS_{ρ} , and $L1-SVM_{\rho}$ solutions compared to the golden reference [5]. For further validation, it includes additional experiments of similar setup using predictive technology models [79], [80]. We report on average an error in the yield sigma estimate of around 0.19 for BBR-OFS with maximum error below 0.5σ . This is compared to an average sigma error of 0.34 and 0.44 for the OFS_{ρ} and $L1-SVM_{\rho}$ with a maximum error of 1.5σ for both. This clearly highlights the advantages of data balancing together with OFS.

1) *Runtime*: From the perspective of runtime complexity, the explicit methods offered clear speedup advantages

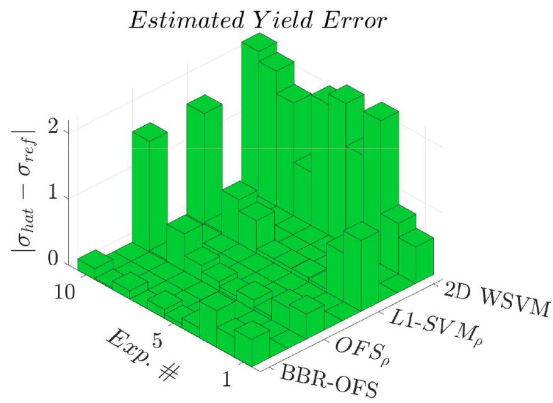


Fig. 11. Yield error, $|\sigma_{\text{err}}| = |\sigma_{\text{hat}} - \sigma_{\text{ref}}|$, where σ_{ref} is the yield for the pure-circuit simulation-based approach. σ_{hat} is the yield corresponding to the model-based importance sample points evaluation for the different approaches.

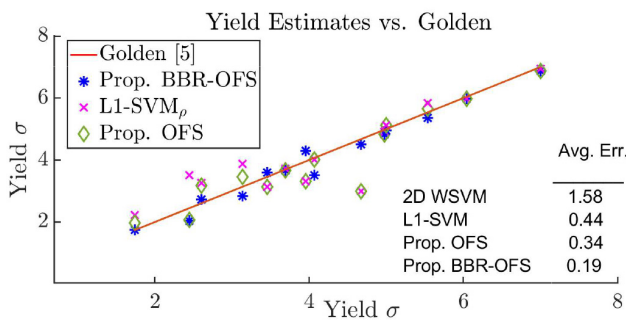


Fig. 12. Scatter yield plot for pure circuit simulation-based importance sampling approach (golden) [5], versus BBR-OFS, OFS, and L1-SVM $_{\rho}$ for the above described above industrial design experiments along with few additional data points of similar experimental setup that rely on ptm [79].

as compared to the pure circuit simulation-based importance sampling approach [5] (see Table I) by eliminating the need for circuit simulations in the importance sampling phase. The model building was performed on the uniform dataset which comprised upto ~ 2000 data points for the balanced datasets depending on the raw balance ratio. The yield estimation was performed on close to 1000 importance sample points on a large memory cross-section. Model building in *R* and SPICE simulations were performed on a 4-GHz IBM Power7 core processor machine. The BBR-OFS methodology demonstrated $\sim 179\times$ speedup for Phase 2 simulations thereby enabling significant speedup compared to the pure circuit simulation-based approach which is much needed when simulating large cross-sections. Speedup for L1-SVM and OFS $_{\rho}$ is also presented and is demonstrated to be higher for Phase 2 simulations due to the fact that BBR-OFS involves searching for the best balance ratios and that SMOTE enlarges the training datasets. As such, for L1-SVM we report 0.65 s of runtime for the model building (fitting) using *R* compared to 19 s for OFS $_{\rho}$ and 150 s for the BBR-OFS. Hence, OFS $_{\rho}$ offers an intermediate solution in terms of on average comparable accuracy to BBR-OFS and model building costs. Nevertheless, considering the overall runtime breakdown that consists of the cost for uniform sampling Phase 1 simulations, and the model building and model evaluations for Phase 2, the model building overhead in BBR-OFS is small and can be justified. This is especially true given that it demonstrated a tighter yield error spread compared to the other methods. Finally, the implicit 2DWSVM-based approach lagged from a runtime perspective due to the cost of

the model-building phase which seemed to suffer in terms of the number of sample points and due to the 2-D space explorations. We also note that the model evaluations cost for all methods was a fraction of a second. In conclusion, we observe that, our proposed BBR-OFS approach presents advantages both in terms of yield accuracy and runtime efficiency in relation to the classical pure-circuit simulation rare-event estimation importance sampling-based approach [5], and, as such, to other similar classical rare event simulators of comparable efficiency to [5], such as [81] and [82].

V. CONCLUSION

We proposed a best balance ratio OFS methodology that aims at boosting classifier performance in the context of data imbalance aware rare fail event estimation. We tackle the OFS problem as a two-level optimization framework approximation to the L_0 -norm regularization. We rely on SMOTE as a popular data balancing method in the ML paradigm for probing for the best balance ratio. We investigate the accuracy and performance tradeoffs of the proposed methodology when estimating the yield of an industrial 14-nm FinFET SRAM design. Our results demonstrate that BBR-OFS outperforms other methods and offers an accurate estimate of the memory yield with an average error of 0.19 σ for the yield estimate and runtime improvement of $179\times$ compared to the pure-circuit simulation-based approach. As part of future work, we plan to explore the feasibility of SMOTE as a potential aid to importance sample point generation.

REFERENCES

- [1] S. Nassif, "Delay variability: Sources, impacts and trends," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2000, pp. 368–369.
- [2] A. Singhee and R. A. Rutenbar, "Statistical blockade: Very fast statistical simulation and modeling of rare circuit events and its application to memory design," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 28, no. 8, pp. 1176–1189, Aug. 2009.
- [3] K. Agarwal and S. Nassif, "Statistical analysis of SRAM cell stability," in *Proc. 43rd ACM/IEEE Design Autom. Conf.*, 2006, pp. 57–62.
- [4] S. Mukhopadhyay, H. Mahmoodi, and K. Roy, "Statistical design and optimization of SRAM cell for yield enhancement," in *Proc. IEEE/ACM Int. Conf. Comput. Aided Design*, 2004, pp. 10–13.
- [5] R. Kanj, R. Joshi, and S. Nassif, "Mixture importance sampling and its application to the analysis of SRAM designs in the presence of rare failure events," in *Proc. 43rd ACM/IEEE Design Autom. Conf.*, 2006, pp. 69–72.
- [6] J. Yao, Z. Ye, and Y. Wang, "Efficient importance sampling for high-sigma yield analysis with adaptive online surrogate modeling," in *Proc. Design, Autom. Test Europe Conf. Exhibit. (DATE)*, 2013, pp. 1291–1296.
- [7] S. Sun, Y. Feng, C. Dong, and X. Li, "Efficient SRAM failure rate prediction via Gibbs sampling," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 31, no. 12, pp. 1831–1844, Dec. 2012.
- [8] L. Dolecek, M. Qazi, D. Shah, and A. Chandrakasan, "Breaking the simulation barrier: SRAM evaluation through norm minimization," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design*, 2008, pp. 322–329.
- [9] I. M. Elfadel, D. S. Boning, and X. Li, *Machine Learning in VLSI Computer-Aided Design*. Cham, Switzerland: Springer, 2019.
- [10] L. Shaer, R. Kanj, R. Joshi, M. Malik, and A. Chehab, "Regularized logistic regression for fast importance sampling based SRAM yield analysis," in *Proc. 18th Int. Symp. Qual. Electron. Design (ISQED)*, 2017, pp. 119–124.
- [11] H. Chen and D. Boning, "Online and incremental machine learning approaches for IC yield improvement," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2017, pp. 786–793.
- [12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002.
- [13] T. Lee, K. B. Lee, and C. O. Kim, "Performance of machine learning algorithms for class-imbalanced process fault detection problems," *IEEE Trans. Semicond. Manuf.*, vol. 29, no. 4, pp. 436–445, Nov. 2016.

- [14] M. Malik, R. V. Joshi, R. Kanj, S. Sun, H. Homayoun, and T. Li, "Sparse regression driven mixture importance sampling for memory design," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 1, pp. 63–72, Jan. 2018.
- [15] L. Shaer, R. Kanj, and R. Joshi, "Data imbalance handling approaches for accurate statistical modeling and yield analysis of memory designs," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2019, pp. 1–5.
- [16] R. Dietrich, M. Oppel, and H. Sompolinsky, "Statistical mechanics of support vector networks," *Phys. Rev. Lett.*, vol. 82, no. 14, p. 2975, 1999.
- [17] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets," in *Proc. Eur. Conf. Mach. Learn.*, 2004, pp. 39–50.
- [18] H. He and Y. Ma, *Imbalanced Learning: Foundations, Algorithms, and Applications*. Hoboken, NJ, USA: Wiley, 2013.
- [19] G. Wu and E. Y. Chang, "Adaptive feature-space conformal transformation for imbalanced-data learning," in *Proc. 20th Int. Conf. Mach. Learn. (ICML)*, 2003, pp. 816–823.
- [20] G. Wu and E. Y. Chang, "Class-boundary alignment for imbalanced dataset learning," in *Proc. Workshop Learn. Imbalanced Data Sets II*, Washington, DC, USA, 2003, pp. 49–56.
- [21] Y. Lin, Y. Lee, and G. Wahba, "Support vector machines for classification in nonstandard situations," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 191–202, 2002.
- [22] K. Veropoulos, C. Campbell, and N. Cristianini, "Controlling the sensitivity of support vector machines," in *Proc. Int. Joint Conf. AI*, 1999, p. 55–60.
- [23] A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, *Learning From Imbalanced Data Sets*, vol. 10. Cham, Switzerland: Springer, 2018.
- [24] A. Kowalczyk and B. Raskutti, "One class SVM for yeast regulation prediction," *ACM SIGKDD Explorations Newslett.*, vol. 4, no. 2, pp. 99–100, 2002.
- [25] S. Ertekin, J. Huang, L. Bottou, and L. Giles, "Learning on the border: Active learning in imbalanced data classification," in *Proc. 16th ACM Conf. Conf. Inf. Knowl. Manage.*, 2007, pp. 127–136.
- [26] G. Wu and E. Y. Chang, "KBA: Kernel boundary alignment considering imbalanced data distribution," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 786–795, Jun. 2005.
- [27] C.-Y. Yang, J.-S. Yang, and J.-J. Wang, "Margin calibration in SVM class-imbalanced learning," *Neurocomputing*, vol. 73, nos. 1–3, pp. 397–411, 2009.
- [28] R. Batuwita and V. Palade, "FSVM-CIL: Fuzzy support vector machines for class imbalance learning," *IEEE Trans. Fuzzy Syst.*, vol. 18, no. 3, pp. 558–571, Jun. 2010.
- [29] V. Palade, "Class imbalance learning methods for support vector machines," in *Imbalanced Learning: Foundations, Algorithms, and Applications*. Hoboken, NJ, USA: Wiley, 2013, p. 83.
- [30] S. J. Shin, Y. Wu, and H. H. Zhang, "Two-dimensional solution surface for weighted support vector machines," *J. Comput. Graph. Stat.*, vol. 23, no. 2, pp. 383–402, 2014. [Online]. Available: <https://doi.org/10.1080/10618600.2012.761139>
- [31] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263–1284, Sep. 2009.
- [32] I. Mani and I. Zhang, "kNN approach to unbalanced data distributions: A case study involving information extraction," in *Proc. Workshop Learn. Imbalanced Datasets*, vol. 126, 2003, pp. 1–7.
- [33] R. Batuwita and V. Palade, "Efficient resampling methods for training support vector machines with imbalanced datasets," in *Proc. Int. Joint Conf. Neural Netw.*, 2010, pp. 1–8.
- [34] Z. Lin, Z. Hao, X. Yang, and X. Liu, "Several SVM ensemble methods integrated with under-sampling for imbalanced data learning," in *Proc. Int. Conf. Adv. Data Min. Appl.*, 2009, pp. 536–544.
- [35] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "SMOTEBoost: Improving prediction of the minority class in boosting," in *Proc. Eur. Conf. Principles Data Min. Knowl. Disc.*, 2003, pp. 107–119.
- [36] G. M. Weiss and F. Provost, "Learning when training data are costly: The effect of class distribution on tree induction," *J. Artif. Intell. Res.*, vol. 19, no. 1, pp. 315–354, 2003.
- [37] I. Albusua, O. Arbelaitz, I. Gurrutxaga, A. Lasarguren, J. Muguerza, and J. M. Pérez, "The quest for the optimal class distribution: An approach for enhancing the effectiveness of learning via resampling methods for imbalanced data sets," *Progr. Artif. Intell.*, vol. 2, no. 1, pp. 45–63, 2013.
- [38] A. Nath and K. Subbiah, "Probing an optimal class distribution for enhancing prediction and feature characterization of plant virus-encoded RNA-silencing suppressors," *3 Biotech*, vol. 6, no. 1, p. 93, 2016.
- [39] J. Yuan, J. Li, and B. Zhang, "Learning concepts from large scale imbalanced data sets using support cluster machines," in *Proc. 14th ACM Int. Conf. Multimedia*, 2006, pp. 441–450.
- [40] N. V. Chawla, "C4.5 and imbalanced data sets: Investigating the effect of sampling method, probabilistic estimate, and decision tree structure," in *Proc. ICML*, vol. 3, 2003, p. 66.
- [41] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik, "Feature selection for SVMs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 668–674.
- [42] S. Perkins, K. Lacker, and J. Theiler, "Grafting: Fast, incremental feature selection by gradient descent in function space," *J. Mach. Learn. Res.*, vol. 3, pp. 1333–1356, Mar. 2003.
- [43] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *J. Mach. Learn. Res.*, vol. 3, pp. 1289–1305, Mar. 2003.
- [44] Z. Zheng, X. Wu, and R. Srihari, "Feature selection for text categorization on imbalanced data," *ACM SIGKDD Explorations Newslett.*, vol. 6, no. 1, pp. 80–89, 2004.
- [45] A. B. Chan, N. Vasconcelos, and G. R. Lanckriet, "Direct convex relaxations of sparse SVM," in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 145–153.
- [46] C. Yi and J. Huang, "Semismooth newton coordinate descent algorithm for elastic-net penalized Huber loss regression and quantile regression," *J. Comput. Graph. Stat.*, vol. 26, no. 3, pp. 547–557, 2017.
- [47] N. Metropolis and S. Ulam, "The Monte Carlo method," *J. Amer. Stat. Assoc.*, vol. 44, no. 247, pp. 335–341, 1949.
- [48] V. N. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY, USA: Springer, 1995.
- [49] O. Chapelle, P. Haffner, and V. N. Vapnik, "Support vector machines for histogram-based image classification," *IEEE Trans. Neural Netw.*, vol. 10, no. 5, pp. 1055–1064, Sep. 1999.
- [50] Z. Zhang, X. Gu, Y. Xie, Z. Wang, Z. Wang, and K. Chakrabarty, "Diagnostic system based on support-vector machines for board-level functional diagnosis," in *Proc. 17th IEEE Eur. Test Symp. (ETS)*, 2012, pp. 1–6.
- [51] A. Ng, "Part V: Support vector machines," Lecture Notes CS229, Stanford Univ., Stanford, CA, USA, 2012.
- [52] S. Maldonado, R. Weber, and J. Basak, "Simultaneous feature selection and classification using kernel-penalized support vector machines," *Inf. Sci.*, vol. 181, no. 1, pp. 115–128, 2011.
- [53] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY, USA: Springer, 2009.
- [54] M. Kuhn and K. Johnson, "An introduction to feature selection," in *Applied Predictive Modeling*. New York, NY, USA: Springer, 2013, pp. 487–519.
- [55] X. Li, "Finding deterministic solution from underdetermined equation: Large-scale performance modeling by least angle regression," in *Proc. 46th ACM/IEEE Design Autom. Conf.*, 2009, pp. 364–369.
- [56] T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu, "The entire regularization path for the support vector machine," *J. Mach. Learn. Res.*, vol. 5, pp. 1391–1415, Oct. 2004.
- [57] J. Zhu, S. Rosset, R. Tibshirani, and T. J. Hastie, "1-norm support vector machines," in *Proc. Adv. Neural Inf. Process. Syst.*, 2003, pp. 49–56.
- [58] J. Bi, K. Bennett, M. Embrechts, C. Breneman, and M. Song, "Dimensionality reduction via sparse support vector machines," *J. Mach. Learn. Res.*, vol. 3, pp. 1229–1243, Mar. 2003.
- [59] S. S. Keerthi, "Generalized LARS as an effective feature selection tool for text classification with SVMs," in *Proc. 22nd Int. Conf. Mach. Learn.*, 2005, pp. 417–424.
- [60] Y. Yang and H. Zou, "An efficient algorithm for computing the HHSVM and its generalizations," *J. Comput. Graph. Stat.*, vol. 22, no. 2, pp. 396–415, 2013.
- [61] C. Yi and Y. Zeng, "SparseSVM: Solution paths of sparse high-dimensional support vector machine with lasso or elastic-net regularization," R package version 1.1-6. 2018. [Online]. Available: <https://CRAN.R-project.org/package=sparseSVM>
- [62] B. Efron and R. J. Tibshirani, *Cross-Validation and the Bootstrap: Estimating the Error Rate of a Prediction Rule*, Division Biostat., Stanford Univ., Stanford, CA, USA, 1995.
- [63] S. Rosset and J. Zhu, "Piecewise linear regularized solution paths," *Ann. Stat.*, vol. 35, no. 3, pp. 1012–1030, 2007.
- [64] L. Wang, J. Zhu, and H. Zou, "Hybrid huberized support vector machines for microarray classification and gene selection," *Bioinformatics*, vol. 24, no. 3, pp. 412–419, 2008.

- [65] L. Demidova and I. Klyueva, "Improving the classification quality of the SVM classifier for the imbalanced Datasets on the base of ideas the SMOTE algorithm," in *Proc. ITM Web Conf.*, vol. 10, 2017, p. 2.
- [66] Y. Zhou, M. Han, L. Liu, J. S. He, and Y. Wang, "Deep learning approach for cyberattack detection," in *Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS)*, 2018, pp. 262–267.
- [67] G. King and L. Zeng, "Logistic regression in rare events data," *Political Anal.*, vol. 9, no. 2, pp. 137–163, 2001.
- [68] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intell. data Anal.*, vol. 6, no. 5, pp. 429–449, 2002.
- [69] R. Sifa, J. Runge, C. Bauckhage, and D. Klapper, "Customer lifetime value prediction in non-contractual freemium settings: Chasing high-value users using deep neural networks and SMOTE," in *Proc. 51st Hawaii Int. Conf. Syst. Sci.*, 2018, pp. 1–10.
- [70] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "DBSMOTE: Density-based synthetic minority over-sampling technique," *Appl. Intell.*, vol. 36, no. 3, pp. 664–684, 2012.
- [71] C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "Safe-level-smote: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem," in *Proc. Pacific-Asia Conf. Knowl. Disc. Data Min.*, 2009, pp. 475–482.
- [72] X. Li, "Finding deterministic solution from underdetermined equation: Large-scale performance variability modeling of analog/RF circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 29, no. 11, pp. 1661–1668, Nov. 2010.
- [73] J. Wang, X. Shen, and Y. Liu, "Probability estimation for large-margin classifiers," *Biometrika*, vol. 95, no. 1, pp. 149–167, 2008.
- [74] R. V. Joshi and M. M. Ziegler, "Programmable supply boosting techniques for near threshold and wide operating voltage SRAM," in *Proc. IEEE Custom Integr. Circuits Conf.*, 2017, pp. 1–4.
- [75] S. J. Shin, *Wsvmurf: Two-Dimensional Solution Surface of the Weighted Support Vector Machine Algorithm, R Package Version 1.0*, 2010.
- [76] N. Rout, D. Mishra, and M. K. Mallick, "Handling imbalanced data: A survey," in *Proc. Int. Adv. Soft Comput. Intell. Syst. Appl.*, 2018, pp. 431–443.
- [77] Y. Chen, Y. Lin, T. Gai, Y. Su, Y. Wei, and D. Z. Pan, "Semisupervised Hotspot detection with self-paced multitask learning," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 39, no. 7, pp. 1511–1523, Jul. 2020.
- [78] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (ROC) curve," *Radiology*, vol. 143, no. 1, pp. 29–36, 1982, doi: 10.1148/radiology.143.1.7063747.
- [79] "Predictive technology model." 2011. [Online]. Available: <http://ptm.asu.edu>
- [80] W. Zhao and Y. Cao, "New generation of predictive technology model for sub-45 nm early design exploration," *IEEE Trans. Electron Devices*, vol. 53, no. 11, pp. 2816–2823, Nov. 2006.
- [81] S. Sun and X. Li, "Fast statistical analysis of rare circuit failure events via subset simulation in high-dimensional variation space," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2014, pp. 324–331.
- [82] M. Rakka, M. E. Fouda, R. Kanj, A. Eltwil, and F. J. Kurdahi, "Design exploration of sensing techniques in 2T-2R resistive ternary CAMs," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 68, no. 2, pp. 762–766, Feb. 2021.



Rouwaida Kanj (Senior Member, IEEE) received the M.S. and Ph.D. degrees in electrical engineering from the University of Illinois Urbana-Champaign, Champaign, IL, USA, in 2000 and 2004, respectively.

She is currently a tenured Associate Professor on leave with the American University of Beirut, Beirut, Lebanon. From 2004 to 2012, she worked as a Research Staff Member with IBM Austin Research Labs, Austin, TX, USA. He joined Synopsys Research and Development Advanced Variational Analysis Group as a member of Technical Staff in 2022. She has authored more than 90 technical papers, 38 issued U.S. patents, and several pending patents. Her research work focuses on advanced algorithmic research and development and smart analytics methodologies for design for manufacturability reliability and yield with emphasis on statistical analysis, optimization, and rare fail event estimation for microprocessor memory designs along with machine learning applications for very large scale integration. More recently she is also involved in memristor-based memory design and reliability and the design of reliable circuits and systems for healthcare and in-memory compute. This is in addition to her earlier work on noise modeling and characterization of CMOS designs.

Dr. Kanj received the Outstanding Technical Achievement Award and six Invention Plateau awards from IBM. She received the prestigious IEEE/ACM WILLIAM J. MCCALLA ICCAD Best Paper Award in 2009, two IEEE ISQED Best Paper Awards in 2006 and 2014, and the IEEE ICM Best Paper Award in 2020. She received the American University of Beirut Teaching Excellence Award in 2021. She was a recipient of three IBM Ph.D. Fellowships. In 2018, her work on statistical yield analysis methodology was nominated for was nominated for the ACM/IEEE Richard Newton Award. She is currently serves or chairs on the technical program committees of several prestigious IEEE conferences. She also serves as a peer-reviewer for many prestigious journals.



Rajiv V. Joshi (Life Fellow, IEEE) received the B.Tech. degree from the Indian Institute of Technology Bombay (IIT Bombay), Mumbai, India, in 1977, the M.S. degree from University of Maine, Maine, Orono, ME, USA, in 1979, the Mech.Eng. degree from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1981, and the Dr.Eng.Sc. degree from Columbia University, New York, NY, USA, in 1990.

He is a Research Staff Member and the Key Technical Lead with T. J. Watson Research Center, IBM, Ossining, NY, USA. He has led successfully predictive failure analytic techniques for yield prediction and also the technology-driven SRAM at IBM Server Group. His statistical techniques are tailored for machine learning and AI. He developed novel memory designs which are universally accepted. He commercialized these techniques. He has authored and coauthored over 220 papers. He holds 62 invention plateaus, and has over 275 U.S. patents and over 415, including international patents.

Dr. Joshi received three Outstanding Technical Achievements, three highest Corporate Patent Portfolio Awards for licensing contributions. He received the NY IP Law Association "Inventor of the year" Award in February 2020. He is awarded the prestigious IEEE Daniel Noble Award Fin 2018 for contributions to predictive analytics, circuits, and technology. He received the Industrial Pioneer Award 2014 from IEEE Circuits and Systems society. He received the Best Editor Award from IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION SYSTEMS. He is inducted into New Jersey Inventor Hall of Fame in August 2014. He is a recipient of the 2013 IEEE CAS Industrial Pioneer Award and the 2013 Mehboob Khan Award from Semiconductor Research Corporation. He won this award again in 2020 for AI initiatives in the BRIC program funded by SRC. He won several best paper awards from ISSCC 1992, ICCAD 2012, ISQED, and VMIC. Also, he has given over 60 invited/keynote talks and given several Seminars. He serves in the Board of Governors for IEEE CAS as an Industrial Liaison. He serves as an Associate Editor of IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION SYSTEMS and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS. He has served on committees of conferences, such as DAC 2019, AICAS 2019, ISCAS, International Symposium Low Power Electronic Design, IEEE VLSI design, IEEE CICC, IEEE International SOI conference, ISQED, and Advanced Metallization. He is an Industry Liaison for universities as a part of the Semiconductor Research Corporation and IEEE CAS society. He is a member of IBM Academy of Technology and a master inventor. He is the ISQED and World Technology Network Fellow and the Distinguished Alumnus of IIT Bombay. He served as a Distinguished Lecturer for IEEE CAS, CEDA, and EDS Society.



Lama Shaer (Student Member, IEEE) received the M.S. and Ph.D. degrees in electrical engineering from the American University of Beirut, Beirut, Lebanon, in 2014 and 2021, respectively.

She is currently an Assistant Manager of Data Scientist with Hyundai Motors America, Fountain Valley, CA, USA. In 2019, she did a six-month internship with BMW Group, Munich, Germany. From 2014 to 2015, she worked as a Systems Engineer with Ublite Inc, San Diego, CA, USA. Her research interests are mainly in machine learning and smart analytics applications to circuit design reliability and rare fail event estimation.

Authorized licensed use limited to: American University of Beirut. Downloaded on March 27, 2024 at 09:36:05 UTC from IEEE Xplore. Restrictions apply.