Semiconductor Manufacturing Final Test Yield Optimization and Wafer Acceptance Test Parameter Inverse Design Using Multi-Objective Optimization Algorithms

DAN JIANG\textsuperscript{1,2}, WEIHUA LIN\textsuperscript{2}, AND NAGARAJAN RAGHAVAN\textsuperscript{1}, (Member, IEEE)

1Engineering Product Development, Singapore University of Technology and Design, Singapore 487372
2Silicon Laboratories International, Singapore 539775
Corresponding author: Dan Jiang (dan_jiang@mymail.sutd.edu.sg)

This work was supported by the Economic Development Board of Singapore (EDB) and Silicon Laboratories, Inc. (SLAB) through the Industry Postgraduate Program (IPP) under Grant IGIPSILICON1901.

ABSTRACT In the semiconductor industry, many previous optimization studies have been carried out at the integrated circuit front-end design phase to identify optimal circuit elements’ size and design parameters. With the scaling of device dimensions, semiconductor manufacturing back-end Final Test (FT) yield is increasingly influenced by systematic or random process variations. As a result, the FT yield is not fully guaranteed by design phase optimization due to existing limitations of yield simulation models and coverage. Very few studies have attempted to incorporate the production FT yield data into process variation optimization and inverse design. In this paper, we introduce a novel framework for FT yield optimization and Wafer Acceptance Test (WAT) parameter inverse design using multi-objective optimization algorithms. This provides a solution to monitor and quickly adjust process variations to maximize FT yield without expensive characterization or design correction after products are released into the production phase. Both yield optimization and process shift feasibility are to be taken into consideration by formulating these factors into a two objective optimization problem. One objective is to minimize FT yield loss, wherein the yield loss is predicted by a machine learning model. The other objective is to minimize the total WAT parameters shift distance from the current standard setting. Three widely used multi-objective algorithms NSGA-II, SMPSO and GDE3 are applied, compared and discussed. An automatic parameter tuning approach using Sequential Model-based Algorithm Configuration (SMAC) and entropy-based termination criterion is applied to reduce the execution time whilst maintaining the optimization algorithms’ performance. Real production data for CMOS 55nm chip are used to validate the framework and the results point to the effectiveness of yield improvement and accurate identification of the optimal WAT parameter combination.

INDEX TERMS Semiconductor manufacturing, yield prediction, inverse design, final test, multi-objective optimization, smart manufacturing, process variation.

I. INTRODUCTION

With the increasing cost of semiconductor fabrication due to technology node shrinking and global supply shortage situations, the manufacturing yield optimization is becoming one of the most critical goals for semiconductor operations. Current production yield improvement strategies mainly rely on engineers’ manual monitoring and hindsight and any corrective actions can only take place after integrated circuit (IC) finished assembly and testing as mentioned in [1]. Immense amount of data are generated during semiconductor manufacturing processes but are not fully utilized. It is important to predict and optimize yield at an earlier manufacturing stage to reduce forgone production losses and scrap-induced shortage. Previous studies on semiconductor yield optimization mainly focus on the IC front-end design instead of using...
production data for yield optimization. The purpose of design phase yield optimization is to avoid expensive redesign and redo the development and manufacturing process chain [2]. The common method for design phase yield optimization is based on yield model simulation using defect distribution and sensitivity. The field of Design for Yield (DFY) deals with a design flow [2]–[4] aimed at coming up with an optimal size for the circuit elements and to adjust the design parameter values to ensure functionality of the circuit and maximize the production yield. For design phase optimization, there are two major areas of work, one is system level hierarchical optimization [5] and the other is the building block level yield optimization [6]. In system level hierarchical optimization, a non-dominated sorting based global optimization algorithm is presented in [7] to find the yield-aware Pareto surfaces for optimal voltage controlled oscillator (VCO) circuit. A yield-aware Pareto surface consisting of points on the Pareto surface guarantees a fixed yield number while meeting system level specifications. In [5], the Kriging model [8] was used for Phase-Locked Loop (PLL) design optimization. During the iterative optimization process, block level Pareto fronts were generated and then matched to the system-level performance. For building block level yield optimization, linearization at the worst-case points method is applied for mismatch analysis and automatic yield optimization of analog integrated circuits in [6]. An optimized design for the operational transconductance amplifier is discussed in [9]. The authors apply simulated annealing as an optimization routine to then minimize the weighted objective with worst case analysis of both parametric manufacturing and operating point variations.

The above-mentioned design phase yield optimization solutions do not utilize mass production data. The main reasons are that the production data is unavailable at the design phase and there are limited resources for design correction after production release. There is a gap between front-end design and manufacturing - metrology results and no continuous yield optimization is done after the device is released to production. With scaling of device dimensions, the effect of process variations on circuit performance is becoming more pronounced [10] and there is an inevitable limitation to the purely design phase yield optimization. Issues such as the limitations of current device models and use of corner-based methods for robust analog sizing are discussed in [11]. The worst case design approach presented in [6], [9] simulate the performances at the worst-case point, which can introduce inherent errors as failure to account for intra-variations for each device would increase the total number of the process variation variables dramatically. Due to time and cost constraints, DFY is only applied to high volume products or typical product lines, whereas limited simulation and characterization are conducted for other low volume product lines. At design phase, only limited characterization yield data are available for analysis, and most of the DFY methods are applied to individual circuit blocks and not to the overall chip. After production release, there are many factors that can cause low yield problems. This includes not only the process variation, but also different package types and updated test coverage etc. Therefore, the production yield is not fully guaranteed by design phase’s yield simulation. In most cases, it is not feasible to redo the characterization or redesign after full production release due to high cost and limited time-frame. Therefore, it is important to have a solution to optimize overall manufacturing yield and quickly fix process variations with dynamic inferences from other production factors, which is the motivation of our work.

In this paper, we propose a novel framework for semiconductor manufacturing Final Test (FT) yield optimization through inverse design for the Wafer Acceptance Test (WAT) parameters. The framework is able to predict and optimize FT yield at Wafer Fabrication (WF) stage, so that the low yield problems can be resolved at much earlier manufacturing stage compared to current practise [1]. WAT parameters are wafer process measurements sampled at the WF stage which directly reflect the process variations. Typical WAT parameters considered include threshold voltage, saturation drain current, on-state resistance and gate capacitance etc. FT is done for packaged chip at the back-end stage. FT yield is one of the most important factors which contribute to manufacturing cost and forgone production losses as mentioned in [1]. FT has the largest test coverage with the number of tests ranging from hundreds to thousands. The overall test failure rate contribute to overall yield for each wafer. A single WAT parameter can affect multiple different tests in FT. Also, a single test in FT can be correlated to multiple WAT parameters at the same time. Given all this inherent complexity, currently, there is no simple way to formulate the relationship between the WAT parameter and the FT yield. The cycle time between WF to FT stage can take up to 8 to 16 weeks. There are many unknown production variations in between including process variations, equipment conditions and human interference etc. The only feasible approach to address this is using heuristics for WAT parameters inverse design.

Past studies [12]–[15] on yield optimization considering manufacturing factors mainly focused only on probe yield and wafer map design. In [12], a single objective optimization algorithm using Differential Evolution (DE) was used to optimize the die size and improve the probe yield. In the work of Kim et al. [15], DE is again applied to determine the chip size and optimize wafer productivity. The limitation of these studies is that through a single objective optimization, only a single Pareto result can be generated in each simulation. In [14], a tree-based model is introduced to maximize the number of gross dies and increase the wafer fabrication throughput. The limitation of this method is that the optimization procedure is time consuming and require engineers’ knowledge to decide on the search areas.

Many studies [16]–[19] proposed probe yield and wafer map failure pattern prediction methods, but they did not provide a feasible realizable solution from a process variation perspective. Kim et al. in [20] introduced equipment-related
variable selection for probe yield prediction. The limitation of their work is that it only covered two process measurements and did not consider the overall yield. The method therefore does not guarantee yield improvement. In the work of Chien et al. [21], the partial least square method is used for WAT parameters’ analysis. However, the proposed approach is not able to directly provide solution for yield improvement. It requires manual review and analysis for any yield enhancement decision to be made. The above-mentioned typical semiconductor design and manufacturing yield optimization studies have been effectively summarized in Table.1 below.

In our previous work [1], [22], FT yield prediction framework using WAT parameters was introduced and feature importance analysis was applied to identify the root cause (WAT parameter) for low yield samples (sub-population). In this work, we will continue the work in [1], [22] to provide actionable solutions for FT yield improvements through WAT parameter inverse design. To the best of our knowledge, this is the first study aimed at improving semiconductor manufacturing FT yield through WAT parameter inverse design by means of using multi-objective optimization algorithms. In our proposed framework here, the optimal WAT process parameters will be automatically generated. Trade-offs between yield improvement and allowable process shift will be considered during the optimization, which helps engineers making proper decisions based on each objective’s relative importance in the practical situation.

II. YIELD OPTIMIZATION AND WAT PARAMETER INVERSE DESIGN
The flow chart for FT yield optimization enabled inverse design for the WAT parameters is shown in Fig.1. The details of the central portion in the flow chart, referred to as “machine learning yield prediction model”, is presented in sufficient detail in our previous work, [22]. In this study, we focus on the right portion - yield optimization and WAT parameter inverse design. A typical semiconductor manufacturing flow consists of four major stages including WF, Wafer Sort (WS), Assembly and FT as shown in the left portion of Fig.1. The time duration between WF and FT can take up to several months. Therefore, it is important to detect FT low yield problem at the WF stage itself for economic reasons and also to ensure robust quality control as mentioned in [1], [22]. At the WF stage, using the production FT yield and WAT data, a machine learning FT yield prediction model is generated and five key WAT parameters are identified based on feature importance analysis. The optimization target is set to achieve an “ideal” 100% FT yield by adjusting the important WAT parameters’ value. While this target is practically unattainable, we are fixing the ceiling of the objective function to 100% to ensure that the machine learning framework rejects any unrealistic cases from the trained model which may yield unnatural values greater than 100%. In practice, there is no physical meaning for a yield value exceeding 100%. Based on evaluations using the generated model, there is a small 1.3% probability for the predicted results to exceed 100% during optimization. Any points in the parameter space that results in a yield exceeding 100% are automatically discarded from our optimization routine and this crossover beyond 100% is only a result of the imperfect nature of the machine learning predictor. In other words, there is no physical meaning to these points. Meanwhile, the wafer process shift needs to be minimized to maintain device functionality and reduce fabrication risk. We formulate our problem into a two-objective optimization problem (one focusing on maximizing FT yield and the other intending to minimize process shift) and solve it using multi-objective optimization algorithms. Following this, the optimal WAT parameter values are generated and engineers can use the information to then adjust process parameters at WF stage for realizable FT yield improvement.

A. PROBLEM DEFINITIONS
Multi-objective algorithm can be used to search for a set of non-dominant solutions as mentioned in [23]. Pareto front is defined as the trade-off surface formed by the set of Pareto-optimal solutions in the search space. We define our problem as a two-objective optimization problem,

\[
\text{minimize } \{f_1(\vec{x}), f_2(\vec{x})\}
\]

where \(R_{\text{vote}}(\vec{x})\) is the voting regressor generated using the top three performing regressors based on the method mentioned in [22]. As the yield loss has to be non-negative, we introduce the below constraint,

\[
\text{subject } \rightarrow \ g_1(\vec{x}) = f_1(\vec{x}) \geq 0
\]

The \(\vec{x}\) search space should be within 3 sigma based on foundry engineers’ experience. From a foundry point of view, it will cause material functionality and device performance instability if the process is shifted by more than 3 sigma. On the contrary, a choice of 1-2 sigma in a mature process is not significant (and too small a design space) for yield improvement purpose. Assuming that there are \(N\) important WAT parameters, we have \(\vec{x} = (x_1, x_i, \ldots, x_N)\). Here, \(x_i\) represents the normalized value of sigma based on definition mentioned in [1], the search space is defined as,

\[
-3.0 \leq x_i \leq 3.0, \quad i \in (1, \ldots, N)
\]

Minimizing the total process shift is the second objective. The Mahalanobis distance (MD) [24] is used to measure the total WAT sigma shift. WAT parameters are high dimensional parameters and correlated with each other. MD takes multivariate co-variances into consideration which is suitable for
TABLE 1. Summary of typical semiconductor yield optimization studies.

| Category                        | Ref No. | First Author and Affiliation                      | Approach / Methodology | Remarks                                |
|---------------------------------|---------|--------------------------------------------------|------------------------|----------------------------------------|
| Design                          | [7]     | S.K. Tiwary (Carnegie Mellon University)        | Genetic Algorithm      | • Computationally expensive            |
|                                 | [5]     | G. Yu (Texas A&M University)                    | Simulated Annealing    | • Optimal solution not guaranteed to accurately match the actual circuit behavior |
|                                 | [6]     | F. Schenke (Technical University Munich)        | Spec-wise linearization| • Worst case design approach can introduce inherent errors |
|                                 | [9]     | T. Mukherjee (Carnegie Mellon University)      | Simulated Annealing    |                                        |
| Building Block Level Optimization| [12]    | J. Kim (Samsung Electronics)                    | Deep Neural Networks   | • Single objective optimization        |
|                                 | [14]    | C. Chien (National Tsing Hua University)         | Differential Evolution | • Computationally expensive            |
|                                 |         |                                                  | Tree-based model       | • Require engineers’ knowledge to decide on the parameter space search areas |
| Wafer Map Optimization          | [21]    | C. Chien (National Tsing Hua University)        | Modified partial least square | • Not a direct solution for yield improvement |
| Probe Yield Enhancement         | [20]    | K. Kim (Pohang University of Science and Technology) | Partial least square   | • Does not consider the overall yield during prediction |
| Probe Yield Prediction          | [21]    | C. Chien (National Tsing Hua University)        | Modified partial least square | • Not a direct solution for yield improvement |
| FT Yield Prediction             | [1]     | D. Jiang (Singapore University of Technology and Design) | Mixed type of manufacturing data pre-processing | • Potential to adjust manufacturing flow based on the yield classification results |
|                                 | [22]    | D. Jiang (Singapore University of Technology and Design) | Robust output labelling | • Enables WAT parameters root cause analysis |

WAT parameter shift measurement. The second objective is then defined as the MD between two observation points \( \tilde{x} \) and \( \tilde{y} \), with the formula expressed as below:

\[
f_2(\tilde{x}) = (\tilde{x} - \tilde{y})^T C^{-1}(\tilde{x} - \tilde{y})
\]

where \( C^{-1} \) is the inverse of the population correlation matrix, and \( \tilde{x} = (x_1, \ldots, x_N)^T \). The distance is to measure the shifted WAT sigma, thus \( \tilde{y} = (0, \ldots, 0)^T \).

B. MULTI-OBJECTIVE OPTIMIZATION ALGORITHMS

1) NSGA-II

Non-dominated Sorting Genetic Algorithm II (NSGA-II) [25] is a widely used generational genetic algorithm. NSGA-II starts with a random parent population initialization. After solution evaluations, solutions are ranked based on fast non-domination ranking method. During the search iterations, the offspring is reproduced with the same population size as the initial population and genetic operations including selection, crossover and mutation are applied. The parent and offspring populations are combined and sorted according to non-domination. The best solutions are selected based on the ranking results and formed into a new population. NSGA-II uses elite policy to preserve good solutions during iterations. And the non-dominated ranking method is introduced for better convergence. The crowded distance density estimation method is introduced to promote non-dominated population diversity [25]. The crowded-comparison operator is used to choose a solution in a lesser crowded region if the two solutions have same non-domination ranks.
The problem search space. For a swarm with algorithm, each particle represents a potential solution in behavior of birds flocking and fish schooling. In the PSO stochastic optimization technique motivated by the social Kennedy and Eberhart in 1995 [32], is a population-based Particle swarm optimization (PSO) algorithm, developed by

2) SMPSO

Particle swarm optimization (PSO) algorithm, developed by Kennedy and Eberhart in 1995 [32], is a population-based stochastic optimization technique motivated by the social behavior of birds flocking and fish schooling. In the PSO algorithm, each particle represents a potential solution in the problem search space. For a swarm with \( P \) particles, at iteration \( t \), the particle \( x_i \) gets updated in its location based on the below formula where:

\[
\tilde{x}_i^t = \tilde{x}_i^{t-1} + \vec{v}_i^t, \quad i = 1, 2, \ldots, P
\]

the velocity vector \( \vec{v}_i^t \) of each particle can be expressed as:

\[
\vec{v}_i^t = w \cdot \vec{v}_i^{t-1} + C_1 \cdot r_1 \cdot (\tilde{x}_p - \tilde{x}_i^{t-1}) + C_2 \cdot r_2 \cdot (\tilde{x}_g - \tilde{x}_i^{t-1})
\]

where \( \delta_j \) is defined as:

\[
\delta_j = \frac{(\text{upper}_j - \text{lower}_j)}{2}
\]

Moreover, SMPSO uses polynomial mutation [35] as a turbulence factor. To further enhance the search capability, an external leaders archive stores the non-dominated solutions. The crowding distance from NSGA-II is applied here to select which particles should remain in this leaders archive.

The Pseudocode of SMPSO can be summarized as follows:

Algorithm 1: Pseudocode of NSGA-II

Result: Best Pareto Front
Initialize random population;
Evaluate solutions;
Non-Domination Ranking;
while not met Termination Criterion do
    Reproduce Offspring Population;
    Genetic Operations;
    Evaluate Solutions;
    Non-Domination Ranking;
    Crowding Distance Estimation and Selection;
end

The simulation results in [25] shows that NSGA-II has more diversified solutions and better convergence towards the true Pareto front compared to two other algorithms: Pareto-archived evolution strategy (PAES) and strength-Pareto EA (SPEA). The main reason we use NSGA-II is that it exhibits better performance for two-objective optimization problems as mentioned in [26]–[31].

2) SMPSO

Particle swarm optimization (PSO) algorithm, developed by Kennedy and Eberhart in 1995 [32], is a population-based stochastic optimization technique motivated by the social behavior of birds flocking and fish schooling. In the PSO algorithm, each particle represents a potential solution in the problem search space. For a swarm with \( P \) particles, at iteration \( t \), the particle \( x_i \) gets updated in its location based on the below formula where:

\[
\tilde{x}_i^t = \tilde{x}_i^{t-1} + \vec{v}_i^t, \quad i = 1, 2, \ldots, P
\]

the velocity vector \( \vec{v}_i^t \) of each particle can be expressed as:

\[
\vec{v}_i^t = w \cdot \vec{v}_i^{t-1} + C_1 \cdot r_1 \cdot (\tilde{x}_p - \tilde{x}_i^{t-1}) + C_2 \cdot r_2 \cdot (\tilde{x}_g - \tilde{x}_i^{t-1})
\]

Here, \( w \) is the inertial weight, and \( 0 \leq r_1, r_2 \leq 1 \), \( \tilde{x}_p \) is the personal best particle and \( \tilde{x}_g \) is the global best particle, and \( C_1 \) and \( C_2 \) are control parameters.

Speed-constrained multi-objective PSO (SMPSO) is a specific multi-objective PSO algorithm introduced in [33]. It proposes a velocity constriction procedure which can effectively resolve swarm explosion problem [34] by limiting the velocity of the particles. The constriction factor developed by [35] is used to control the particle’s velocity.

To bound the accumulated velocity of each variable \( j \) (in each particle), a velocity constriction equation is defined as follows:

\[
v_{i,j}^t = \begin{cases} 
\text{delta}_j & \text{if } v_{i,j}(t) > \text{delta}_j \\
-\text{delta}_j & \text{if } v_{i,j}(t) \leq -\text{delta}_j \\
v_{i,j}^{t-1} & \text{otherwise}
\end{cases}
\]

Based on the experiments by Nebro et al. in [33], SMPSO outperforms other five state-of-the-art multi-objective optimization algorithms in a typical study in terms of the quality of the Pareto front approximations. We use SMPSO in this work owing to its fast convergence speed and simple operators, and wide usage in different areas to solve practical problems [36]–[38].

3) GDE3

Differential Evolution (DE) is another popular optimization approach which was first introduced by Storn and Price in 1995 [39]. The DE algorithm includes random population initialization, selection, mutation, and crossover operations. Generalized Differential Evolution (GDE) extended DE with a modified selection rule based on constraint-domination [40]. GDE shows good performance for constrained multi-objective optimization problems but it is sensitive to the control parameters [41]. Kukkonen and Lampinen proposed a third version of GDE, GDE3, which is an extension of DE for global optimization with an arbitrary number of objectives and constraints [42]. GDE3 modifies the earlier GDE versions using a growing population and non-dominated sorting to decrease the population size at the end of each generation as mentioned in [42]. It also introduces a crowding distance to approximate the crowdedness of a vector in its non-dominated set. The crowding distance is improved over NSGA-II to provide better distributed solutions. Non-dominated sorting is also modified to consider constraints. The experimental results in [42] show that GDE3 solutions are better distributed and more stable compared to its previous versions viz. of GDE and NSGA-II.
GDE3 demonstrates good performance in terms of diversity, especially for two-objective optimization problems, and furthermore, it is found to be efficient in solving constrained multi-objective problems [43].

Algorithm 3: Pseudocode of GDE3

Result: Best solution population
Initialize random population;
Evaluate solutions;
while not met Termination Criterion do
    Reproduce offspring population;
    Differential Evolution Selection;
    Differential Evolution Crossover;
    Evaluate Solutions;
    Non-Domination Sorting;
    Crowding Distance Estimation and Selection;
end

C. PERFORMANCE INDICATOR

Performance indicator is a measurement metric used to compare the performance of different multi-objective optimization algorithms. One widely used indicator is Generational Distance (GD), which measures the closeness of non-dominated solutions to the reference Pareto front set [44]. The Inverted Generational Distance (IGD) is another commonly used indicator proposed in [45], which is an improved version of GD. However, both GD and IGD require the knowledge of reference Pareto front set which is not suitable for our optimization problem. All data used in this study are real world manufacturing data. In the context of our work, the reference Pareto front set is unavailable at the point of WAT and yield optimization. In typical physics, chemical, economic or financial domains, it is easy to define the reference Pareto front based on statistical models. In our study, the Pareto front set is explored using a machine learning model with inherent prediction errors. It is therefore not feasible to define the reference outcome of the machine learning model.

Considering this scenario, the Hypervolume (HV) indicator is used in this study. The HV indicator was first presented by Zitzler and Thiele in [46]. HV has been the most commonly used performance indicator for multi-objective algorithms due to its strict compliance with the Pareto dominance [47] and its provision for a comprehensive quality measurement. HV calculation only requires a reference point instead of a reference Pareto front set. Given a solution set \( S \) and a reference point \( r \), the HV is defined as

\[
HV(S) = \text{Leb}(\bigcup_{x \in S} [x|s < x < r])
\]

(10)

where \( \text{Leb} \) denotes the Lebesgue measure. Therefore, HV value of a solution set \( S \) represents the volume of the union of the hypercubes determined by each of its solutions and the reference point \( r \) as described in [48].

Since the reference point will strongly affect the HV result, it is important to properly choose the reference point for a fair algorithm comparison. Consider a normalized objective space with the ideal point \((0, 0, \ldots, 0)\) and the nadir point \((1, 1, \ldots, 1)\). Let \( \mu \) be the number of solutions. One of the suggested two-objective optimization reference point mentioned in [49] is as follows:

\[
r = 1 + \frac{1}{\mu - 1}
\]

(11)

In our two-objective optimization case here, a higher HV value indicates a better solution. The reference point is the one at the maximum yield loss and maximum WAT sigma shift distance. The actual calculation for the HV used in our work is based on the work of Fonseca et al. [50].

III. FAB PRODUCTION DATA, ANALYSIS RESULTS AND DISCUSSION

A. FAB PRODUCT LINE DETAILS

One CMOS 55nm technology product’s production data is used for the analysis here. All production data are provided by Silicon Laboratories International [50]. The dataset consists of 412 backend lots with FT yield values ranging from 57.6% to 97.1%. Each wafer has 120 numeric WAT parameters. After data pre-processing based on the method mentioned in [22], the number of WAT parameters reduced from 120 to 73. Furthermore, after feature selection, the number of WAT parameters further reduced to 21. The yield loss objective function is based on the FT yield model generated using our previous method introduced in [22]. We explored the performance of six regression models. A voting regressor is generated using the top three models: XGBoost [51], AdaBoost [52] and K-Nearest Neighbor [53]. The most important five WAT parameters are selected based on the feature importance analysis and they are found to be \( \text{Cap}_1, \text{Idsat}_2, \text{Idsat}_4, \text{R}_P, \text{VT}_2 \). The sequence is based on feature importance ranking, where \( \text{Cap}_1 \) is the most important WAT parameter and \( \text{VT}_2 \) is the least important WAT parameter.

B. INITIALIZATION OF OPTIMIZATION ROUTINES

The three optimization algorithms elaborated in the previous section are each run for 30 times independently with a cap of 25000 evaluations as a simple brute force termination criterion. The optimization population/particle size is set as 100 for NSGA-II, SMPSO and GDE3. The NSGA-II algorithm’s parameter setting follows the specifics mentioned in [25]. A simulated binary crossover (SBX) operator is used with a crossover probability set to 0.9. The mutation probability is one divided by the number of decision variables and the distribution index value is set to 20. The SMPSO algorithm follows the settings in [33] with an archive size of 100 while the mutation setting is the same as that of NSGA-II. For the GDE3 algorithm, two control parameters crossover control parameter \( CR \), and the mutation factor \( F \) are set, respectively, to 0.5 and 0.5 based on the reports in [54].
which is the same finding as in [33].

Although SMPSO HV results are the worst compared to other two algorithms, their execution time is the shortest. The NSGA-II and GDE3 results are slightly better than that of SMPSO, while SMPSO results are more widely spread.

To further compare the performance of the algorithms, the HV and execution time results for the 30 runs are presented in the form of a box plot in Fig.3. To compute the HV results, all the computed HV values are normalized to between 0 and 1. The reference point is calculated based on the method described in Section.II-A. From Fig.3, NSGA-II has the best Pareto front among the three algorithms with highest HV results. Also, GDE3 and NSGA-II HV results are found to be more stable and consistent compared to SMPSO because SMPSO solutions are more diversified due to its enhanced exploration capabilities as mentioned in [33]. Although SMPSO HV results are the worst compared to other two algorithms, their execution time is the shortest which is the same finding as in [33].

The simulations are executed on a high performance desktop using Intel I5 10600KF CPU comprising 6 Cores and 12 Threads with 4.10 GHz speed and 16GB RAM.

C. PARETO FRONT RESULTS

The Pareto front for the three algorithms for one of the runs is shown in Fig.2. To validate the WAT optimization results, we intentionally fabricated production lots with shifted WAT parameters. The black crosses in Fig.2 represent those specific production lots which tested with a FT yield loss of less than 3%. When the total WAT sigma shift MD is set to zero, the predicted FT yield loss is 10.7%. It can be seen that all three algorithms’ Pareto fronts are superior to all the production lots data points, highlighting the effectiveness of our yield optimization and WAT inverse design framework here. The NSGA-II and GDE3 results are slightly better than that of SMPSO, while SMPSO results are more widely spread.

To compare the convergence performance of the algorithms, the evolution of HV values over 25000 evaluations in 30 independent runs are displayed in Fig.5. It can be seen that all three algorithms have similar behavior and are able to converge after around 5000 evaluations. To compare the convergence speed, we compute the median and the IQR of the number of evaluations required by the three algorithms to attain 98% the maximum HV.

| Evaluations # | NSGA-II | SMPSO | GDE3 |
|---------------|---------|-------|------|
| Median        | 2.60E+03 | 3.35E+03 | 3.60E+03 |
| IQR           | 3.30E+02 | 5.30E+02 | 3.30E+02 |

To validate whether the proposed framework is able to identify the optimal WAT parameter values, we compare the WAT data of historical production lots and reference production lots, together with the NSGA-II results. Because of the complexity of wafer manufacturing process, we cannot guarantee that a single fabricated lot’s WAT data exactly matches with the targeted NSGA-II results. Therefore, the below discussion focuses on the overall WAT adjustment direction and the distribution of NSGA-II results compared to the reference lots. The historical 6 months’ production lots were fabricated with standard WAT parameters. Their WAT parameters’ mean values are indicated by dot lines in Fig.4. The green area represents the Interquartile Range (IQR) for historical production lots’ WAT data. The IQR is defined as the range between lower 25% percentile to upper 75% percentile. From the NSGA-II results, a total of 51 evaluations from the 30 runs resulted in a yield loss between 1.65% – 1.75%. The 51 evaluations are indicated by the blue color histogram in Fig.4, where the dash lines represents the mean values. From the Fig.4 (a) to (c), we can see that the NSGA-II results suggest that Cap_1 need to be increased, while Idsat_2 and Idsat_4 need to be decreased compared to historical WAT data. Since one of our optimization objective is to also minimize the process shift, the NSGA-II results only suggest a slight increase of R_P1 and nearly no change of VT_2 as indicated in Fig.4 (d) and (e). In synchrony with our feature importance analysis, the results shown in Fig.4 prove that our framework is able to effectively find the optimal WAT parameters’ range for FT yield improvement.

D. CONVERGENCE SPEED

To compare the convergence performance of the three algorithms, the evolution of HV values over 25000 evaluations in 30 independent runs are displayed in Fig.5. It can be seen that all three algorithms have similar behavior and are able to converge after around 5000 evaluations. To compare the convergence speed, we compute the median and the IQR of the number of evaluations required by the three algorithms to attain 98% the maximum HV.
reach 98% the reference HV value. As the true Pareto front is unavailable in our case, the maximum HV value obtained by using NSGA-II is used as the reference. The maximum HV attained is 0.8701.

From Table.2, it is clearly evident that NSGA-II is the fastest in reaching the target HV value. Notice from the inset of Fig.5 that SMPSO is the fastest to reach higher HV during the early phase of evolution. Besides, SMPSO has the shortest total execution time from all the 30 runs, as plotted earlier in Fig.3. If we need to minimize the total optimization time, we can consider using the SMPSO algorithm with a lower percent value of HV as the termination criterion. However, the maximum HV value is unknown before conducting the optimization in practise. In the next section, we discuss on how to reduce the execution time while maintaining the HV performance.

**E. EXECUTION TIME REDUCTION**

In the above experiments, only default parameters’ settings are used as mentioned in Section III-B. In order to maintain and improve HV performance when using early termination criterion, the three algorithms’ parameters are optimized by using an automatic parameter tuning approach called Sequential Model-based Algorithm Configuration (SMAC) [55]. We used the Python implementation developed in [56].

It consists of Bayesian Optimization with an aggressive racing mechanism to efficiently choose better algorithms’ parameters configurations. SMAC provides a good solution at low computational costs for practical problems, and it shows statistically significant improvements over the compared approaches as mentioned in [57]. Other popular automatic tuning criterion like I-Race [58], are not primarily designed for reducing computation time [58]. Traditional parameter tuning criteria including Taguchi [59] and Response Surface Method [60] do not show robust speed and performance for high dimensional optimization problems [61], [62].

The parameters’ configuration space is determined based on past studies in [54], [63]–[65]. The configuration space for population or swarm size is [50, 500] in steps of 50, distribution index is [10, 50] in steps of 10, crossover probability is [0.5, 1.0] in steps of 0.1, the control parameters and mutation factor are [0.0, 1.0] and [0.1, 1.0] in steps of 0.1. The optimization results are shown in Table.3. These optimized parameter settings can be reused for future WAT optimization tasks as well.

For execution time reduction purpose, an entropy-based termination criterion [66] is applied. This termination criterion uses entropy-based dissimilarity measure that helps to determine the stability of several generations of algorithms [66]. Based on the experiments in [66], this criterion obtained faster and qualitatively better approximation at earlier termination for a class of many-objective optimization problems. The main reason we use this termination criterion is that it can effectively prevent a premature termination and avoid wastage of computational resource. Besides, this criterion is able to trigger the termination as well when an optimization falls into dilemma [67]. The basic idea of this termination criterion is to monitor the mean and standard deviation of the dissimilarity measures in a predefined number of successive generations $n_s$. If the mean and standard deviation remain unchanged when comparing the predefined number of decimal places $n_p$ for the $n_s$ generations, then the algorithm can be terminated. The mean $M_t$ and standard deviation $S_t$ at current generation $t$ is defined as follows:

$$M_t = \begin{cases} D_1 & t = 1 \\ \frac{1}{t} \sum_{i=1}^{t} D_i & t \geq 2 \end{cases}$$

(12)

$$S_t = \frac{1}{t} \sum_{i=1}^{t} (D_i - M_t)^2.$$  

(13)

where $i$ represents the generation counter, $D_i$ is the dissimilarity measure at the $i^{th}$ generation.

We rerun the experiment using entropy-based termination criterion with $n_s = 20$ and $n_p = 1$, and applied the optimized parameter settings. The average execution time results over 30 independent runs are shown in Fig.6. It can be seen that the execution time is reduced by 76%, 86%, and 73% for GDE3, SMPSO and NSGA-II compared to the brute force termination criterion and default parameter settings discussed earlier. The HV performance is also maintained for

| Algorithms | Parameters | Optimized Parameters |
|------------|------------|----------------------|
| NSGA-II    | Population size | 100 |
|            | Mutation distribution index | 50 |
|            | Crossover distribution index | 30 |
|            | Crossover probability | 0.6 |
| SMPSO      | Swarm size | 150 |
|            | Archive size | 150 |
|            | Mutation distribution index | 30 |
| GDE3       | Population size | 100 |
|            | Control parameter | 0.5 |
|            | Mutation factor | 0.2 |

**TABLE 3.** Optimized parameters for NSGA-II, SMPSO and GDE3 through Sequential Model-based Algorithm Configuration (SMAC).
GDE3, SMPSO and NSGA-II, by achieving 99.5%, 97.9%, 99.6% of maximum HV based on the averaged results over 30 runs.

IV. CONCLUSION

In this paper, we introduced a novel framework to optimize semiconductor manufacturing FT yield through WAT parameters’ inverse design. This paper serves as a practical and relevant extension of our previous studies in [1], [22], providing an automated procedure to identify the optimal WAT parameter values and considering the trade-offs between yield optimization and process stability by using a multi-objective optimization framework. This framework provides a
comprehensive solution for foundry engineers to monitor and resolve low yield problems at a much earlier manufacturing stage compared to current practice. The FT yield prediction model is able to pre-alert low yield problems at the wafer fabrication stage itself. By using this framework, there is no need for manual review of manufacturing data and no additional resource required from the foundry team; the generated solution can be directly applied to the manufacturing process. To the best of the authors’ knowledge, this is the first work that formulates the FT yield improvement task into a two-objective co-optimization problem using a WAT parameter inverse design approach.

Based on the model assisted low yield root cause analysis, optimization is executed for the sensitive WAT parameters to achieve the highest yield. The resulting Pareto front provides trade-offs between yield improvement and process shift, allowing for easier identification of the preferred optimal solutions based on the objectives’ relative importance. One relatively immature CMOS product line data was used to verify the procedure and the results clearly point to the effectiveness of our model framework in identifying the optimal WAT parameters values for yield enhancement. In our simulation experiment, three widely used multi-objective optimization algorithms, NSGA-II, SMPSO and GDE3 were applied and their outcomes compared with real production lot data.

An automatic parameter tuning approach using SMAC and entropy-based termination criterion are applied to reduce the execution time while maintaining the HV performance. The results show significant execution time reduction, with SMPSO being the most efficient. This can be further reduced through solution evaluation time reduction. The solution evaluation (FT yield prediction from machine learning voting regressor model) step is the major factor contributing to the long execution time. Therefore, it is necessary to investigate how the FT yield model’s prediction time can be significantly reduced. The limitation of our work is that only one 55nm CMOS product line is analyzed, more 55nm product lines and other technology nodes including 40nm and 90nm need to be explored to validate the robustness of the proposed framework.

In the future, we intend to extend the framework to address other complex yield problems, making the framework a one-stop solution for optimizing the entire semiconductor manufacturing line yield in a foundry. For example, the optimization target can be changed to the composite yield including both probe yield and FT yield. The inverse design target should not only be limited to the WAT parameters but should also consider the in-line electric tests’ parameters such as dopant density, spreading resistance and gate poly width etc. In-line test are performed during wafer build, which is generated earlier than WAT parameters. We can predict and resolve low yield problems at an earlier production stage by using in-line parameters as the input data. Besides, categorical parameters including equipment configurations, test facilities, product specifications etc. can also be considered in the inverse design process to make it more holistic and truly applicable to a practical foundry context with multiple vendor process equipments and multiple low volume, varied application product lines.

REFERENCES

[1] D. Jiang, W. Lin, and N. Raghavan. “A novel framework for semiconductor manufacturing final test yield classification using machine learning techniques,” IEEE Access, vol. 8, pp. 197885–197895, 2020.

[2] A. Ripp, M. Buhler, J. Koehl, J. Bickford, J. Hibbeler, U. Schlichtmann, R. Sommer, and M. Pronath, “DATE 2006 special session: DFM/DFY design for manufacturability and yield—influence of process variations in digital, analog and mixed-signal circuit design,” in Proc. Design Automat. Test Eur. Conf., Mar. 2006, pp. 387–392.

[3] S. V. Raman, P. Limpisathan, P. Meinerzhagen, S. Natarajan, and E. Yang, “Automating design for yield: Silicon learning to predictive models and design optimization,” in Proc. IEEE Int. Test Conf. (ITC), Nov. 2020, pp. 1–10.

[4] C. Chiang and J. Kawa, Design for Manufacturability and Yield for Nanoscale CMOS, Dordrecht, The Netherlands: Springer, 2007.

[5] G. Yu and P. Li., “Yield-aware hierarchical optimization of large analog integrated circuits,” in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design, Nov. 2008, pp. 79–84.

[6] F. Schenkel, M. Pronath, S. Zizza, R. Schwenckner, H. Graeb, and K. Antreich, “Mismatch analysis and direct yield optimization by specwise linearization and feasibility-guided search,” in Proc. 38th Conf. Design Automat. (DAC), 2001, pp. 858–863.

[7] S. K. Tiwary, P. K. Tiwary, and R. A. Rutenbar, “Generation of yield-aware Pareto surfaces for hierarchical circuit design space exploration,” in Proc. 43rd Annul. Conf. Design Automat. (DAC), 2006, pp. 31–36.

[8] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, “Design and analysis of computer experiments,” Stat. Sci., vol. 4, no. 4, pp. 409–423, 1989.

[9] T. Mukherjee, L. R. Carley, and R. A. Rutenbar, “Efficient handling of operating range and manufacturing line variations in analog cell synthesis,” IEEE Trans. Comput.-Aided Design Int. Circuits Syst., vol. 19, no. 8, pp. 825–839, Aug. 2000.

[10] A. Agrawal, B. C. Paul, S. Mukhopadhyay, and K. Roy, “Process variation in embedded memories: Failure analysis and variation aware architecture,” IEEE J. Solid-State Circuits, vol. 40, no. 9, pp. 1804–1814, Sep. 2005.

[11] H. E. Graeb, Analog Design Centering and Sizing, vol. 64. Dordrecht, The Netherlands: Springer, 2007.

[12] J.-S. Kim, S.-J. Jang, T.-W. Kim, H.-J. Lee, and J.-B. Lee, “A productivity-oriented wafer map optimization using yield model based on machine learning,” IEEE Trans. Semicond. Manuf., vol. 32, no. 1, pp. 39–47, Feb. 2019.

[13] S.-J. Jang, J.-H. Lee, T.-W. Kim, J.-S. Kim, H.-J. Lee, and J.-B. Lee, “A wafer map yield model based on deep learning for wafer productivity enhancement,” in Proc. 29th Int. SEMI Adv. Semiconductor Manuf. Conf. (ASMC), Apr. 2018, pp. 29–34.

[14] C.-F. Chien and C.-Y. Hsu, “Data mining for optimizing IC feature designs to enhance overall wafer effectiveness,” IEEE Trans. Semicond. Manuf., vol. 27, no. 1, pp. 71–82, Feb. 2014.

[15] J.-S. Kim, C. W. Ahn, T.-W. Kim, H.-J. Lee, and J.-B. Lee, “A market-oriented wafer map optimization methodology using differential evolution to maximize wafer productivity,” in Proc. 28th Annul. SEMI Adv. Semiconductor Manuf. Conf. (ASMC), May 2017, pp. 393–398.

[16] K. Nakata, R. Oritbura, Y. Mizuoka, and K. Takagi, “A comprehensive big-data-based monitoring system for yield enhancement in semiconductor manufacturing,” IEEE Trans. Semicond. Manuf., vol. 30, no. 4, pp. 339–344, Nov. 2017.

[17] J. Wang, Z. Yang, J. Zhang, Q. Zhang, and W.-T. K. Chien, “AdaBalGAN: An improved generative adversarial network with imbalanced learning for wafer defect pattern recognition,” IEEE Trans. Semicond. Manuf., vol. 32, no. 3, pp. 310–319, Aug. 2019.

[18] T. Ishida, I. Nitta, F. Fukuda, and Y. Kanazawa, “Deep learning-based wafer-map failure pattern recognition framework,” in Proc. 20th Int. Symp. Qual. Electron. Design (ISeQED), Mar. 2019, pp. 291–297.

[19] Y. Kong and D. Ni, “A practical yield prediction approach using inline defect metrology data for system-on-chip integrated circuits,” in Proc. 13th IEEE Conf. Automat. Sci. Eng. (CASE), Aug. 2017, pp. 744–749.

[20] K.-J. Kim, K.-J. Kim, C.-H. Jun, L.-G. Chong, and G.-Y. Song, “Variable selection under missing values and unlabeled data in semiconductor processes,” IEEE Trans. Semicond. Manuf., vol. 32, no. 1, pp. 121–128, Feb. 2019.
S. Kukkonen and C. A. C. Coello, “Generalized differential evolution for numerical and evolutionary optimization,” in *NEO*. Cham, Switzerland: Springer, 2017, pp. 253–279.

D. A. V. Veldhuizen and G. B. Lamont, “Evolutionary computation and convergence to a Pareto front,” in *Proc. Late Breaking Papers Genetic Programming Conf.*, 1998, pp. 222–228.

C. A. C. Coello and M. R. Sierra, “A study of the parallelization of a coevolutionary multi-objective evolutionary algorithm,” in *Proc. Mex. Int. Conf. Artif. Intell. Berlin, Germany: Springer, 2004*, pp. 688–697.

E. Zitzler and L. Thiele, “Multiobjective optimization using evolutionary algorithms—a comparative case study,” in *Proc. Int. Conf. Parallel Problem Solving From Nature*. Berlin, Germany: Springer, 1998, pp. 292–301.

E. Zitzler, L. Thiele, M. Laumanns, C. A. C. Coello, and V. G. D. Fonseca, “Performance assessment of multiobjective optimizers: An analysis and review,” *IEEE Trans. Evol. Comput.*, vol. 7, no. 2, pp. 117–132, Apr. 2003.

M. Li and X. Yao, “Quality evaluation of solution sets in multiobjective optimisation: A survey,” *ACM Comput. Surv.*, vol. 52, no. 2, pp. 1–38, May 2019.

H. Ishibuchi, R. Imada, Y. Setoguchi, and Y. Nojima, “How to specify a reference point in hypervolume calculation for fair performance comparison,” *Evol. Comput.*, vol. 26, no. 3, pp. 411–440, Sep. 2018.

C. M. Fonseca, L. Paquette, and M. López-Ibáñez, “An improved dimension-sweep algorithm for the hypervolume indicator,” in *Proc. Int. Conf. Evol. Comput.*, Jul. 2006, pp. 1157–1163.

E. Zitzler, C. A. C. Coello, and L. Thiele, “XGBoost: A scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 785–794.

B. Kégel, “The return of AdaBoost.MH: Multi-class Hamming trees,” 2013, arXiv:1312.6086 [Online]. Available: https://arxiv.org/abs/1312.6086

L. E. Peterson, “K-nearest neighbor,” *Scholarpedia*, vol. 4, no. 2, pp. 1883, 2009.

M. Kukkonen and J. Lampinen, “An empirical study of control parameters for the third version of generalized differential evolution (GDE3),” in *Proc. Int. Conf. Evol. Comput.*, Jul. 2006, pp. 2002–2009.

F. Hutter, H. H. Hoos, and K. Leyton-Brown, “Sequential-model-based optimization for general algorithm configuration,” in *Proc. Int. Conf. Learn. Intell. Optim.*, Berlin, Germany: Springer, 2011, pp. 507–523.

M. Lindauer, K. Eggensperger, M. Fuerer, S. Falkner, A. Biedenkapp, and F. Hutter, (2017). *SMAC v3: Algorithm Configuration in Python.* [Online]. Available: https://github.com/automl/SMAC3

C. Huang, Y. Li, and X. Yao, “A survey of automatic parameter tuning methods for metaheuristics,” *IEEE Trans. Evol. Comput.*, vol. 24, no. 2, pp. 201–216, Apr. 2020.

M. López-Ibáñez, J. Dubois-Lacoste, L. P. Caceres, T. Stützle, and M. Branda, “The ‘race’ package: Iterated racing for automatic algorithm configuration,” *Oper. Res. Perspect.*, vol. 3, pp. 43–58, Jan. 2016.

W. H. Yang and Y. S. Tang, “Design optimization of cutting parameters for turning operations based on the Taguchi method,” *J. Mater. Process. Technol.*, vol. 84, nos. 1–3, pp. 122–129, Dec. 1998.

M. A. Bezerra, R. E. Santelli, E. P. Oliveira, L. S. Villar, and L. A. Escalera, “Response surface methodology (RSM) as a tool for optimization in analytical chemistry,” *Talanta*, vol. 76, no. 5, pp. 965–977, Sep. 2008.

A. E. Eiben and S. K. Smit, “Parameter tuning for configuring and analyzing evolutionary algorithms,” *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 19–31, Mar. 2011.

R. Vitali, *Response Surface Methods for High-Dimensional Structural Design Problems*. Gainesville, FL, USA: Univ. Florida, 2000.

A. Hassanat, K. Almohammad, E. Alkaeween, E. Abunas, A. Hammouri, and V. B. S. Prasath, “Choosing mutation and crossover ratios for genetic algorithms—A review with a new dynamic approach,” *Information*, vol. 10, no. 12, p. 390, Dec. 2019.

M. M. Hamdan, “Revisiting the distribution index in simulated binary crossover,” in *Proc. 3rd Int. Conf. Digital Inf. Commun. Technol.* (DIC-TAP), May 2014, pp. 37–41.

Q. Wang, L. Wang, W. Huang, Z. Wang, S. Liu, and D. A. Savic, “Parameterization of NSGA-II for the optimal design of water distribution systems,” *Water*, vol. 11, no. 5, p. 971, May 2019.

D. K. Saxena, A. Sinha, J. A. Duro, and Q. Zhang, “Entropy-based termination criterion for multiobjective evolutionary algorithms,” *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 485–498, Aug. 2016.

M. Gong, Z. Wang, Z. Zhi, and L. Jiao, “A similarity-based multiobjective evolutionary algorithm for deployment optimization of near space communication system,” *IEEE Trans. Evol. Comput.*, vol. 21, no. 6, pp. 878–897, Dec. 2017.
DAN JIANG received the B.Eng. degree in electrical and electronics engineering and the M.Eng. degree in communications engineering from Nanyang Technological University (NTU), Singapore, in 2013 and 2017, respectively. She is currently pursuing the Ph.D. degree with the Engineering Product Development (EPD) pillar, Singapore University of Technology and Design (SUTD), under the Industry Postgraduate Program (IPP), supported by the Economic Development Board (EDB) of Singapore. Since 2013, she has been working as a Product Test Engineer at Silicon Laboratories International on semiconductor product test and data analytics tools development. Her current research interests include semiconductor manufacturing yield prediction as well as big data and artificial intelligence for semiconductor process and quality optimization.

WEIHUA LIN received the B.Eng. degree in nuclear electronics from the University of Science and Technology of China (USTC), in 1991, and the Master of Science degree in electronics engineering from the National University of Singapore (NUS), in 2001. He is currently the Senior Product Test Engineering (PTE) Director of Silicon Laboratories International. He has close to 30 years of working experience in several semiconductor companies, such as National Semiconductor, Lucent Microelectronics, and Silicon Labs. He has been focusing on integrated circuit (IC) test and product engineering as well as field application and customer support. He is also providing technical consultation roles in IC and module test development, product qualification and yield optimization to achieve good product quality at lower possible cost.

NAGARAJAN RAGHAVAN (Member, IEEE) received the Ph.D. degree in microelectronics from the Division of Microelectronics, Nanyang Technological University (NTU), Singapore, in 2012. He was a Postdoctoral Fellow with the Massachusetts Institute of Technology (MIT), Boston, MA, USA, and at IMEC, Belgium, in joint association with the Katholieke Universiteit Leuven (KUL). He is currently an Assistant Professor with the Engineering Product Development (EPD) pillar, Singapore University of Technology and Design (SUTD). To date, he has authored/coauthored more than 220 international peer-reviewed publications and five invited book chapters as well. His work focuses on reliability assessment, characterization and lifetime prediction of nanoelectronic devices as well as material design for reliability, physics informed machine learning, uncertainty quantification and prognostics and health management of electronic systems. From 2012 to 2014, he was an Invited Member of the IEEE GOLD Committee. He was a recipient of the IEEE Electron Device Society (EDS) Early Career Award for 2016, the Asia-Pacific recipient for the IEEE EDS Ph.D. Student Fellowship in 2011, and the IEEE Reliability Society Graduate Scholarship Award in 2008. He serves as the General Chair for IEEE IPFA 2021 at Singapore and has consistently served on the Review Committee for various IEEE journals and conferences, including IRPS, IIRW, IPFA, and ESREF. He is also serving as an Associate Editor for IEEE Access.

* * *