

Machine Learning for Test, Diagnosis, Post-Silicon Validation and Yield Optimization

*Original*

Machine Learning for Test, Diagnosis, Post-Silicon Validation and Yield Optimization / Amrouch, Hussam; Chakrabarty, Krishnendu; Pflueger, Dirk; Polian, Ilia; Sauer, Matthias; Reorda, Matteo Sonza. - (2022), pp. 1-6. (Intervento presentato al convegno IEEE European Test Symposium tenutosi a Barcelona (ESP) nel 23-27 May 2022) [10.1109/ETS54262.2022.9810416].

*Availability:*

This version is available at: 11583/2981819 since: 2023-09-08T16:13:01Z

*Publisher:*

IEEE

*Published*

DOI:10.1109/ETS54262.2022.9810416

*Terms of use:*

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

*Publisher copyright*

IEEE postprint/Author's Accepted Manuscript

©2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

# Machine Learning for Test, Diagnosis, Post-Silicon Validation and Yield Optimization

Hussam Amrouch\*, Krishnendu Chakrabarty<sup>†</sup>, Dirk Pflüger\*, Ilia Polian\*, Matthias Sauer<sup>‡</sup>, Matteo Sonza Reorda<sup>§</sup>

\*University of Stuttgart, Stuttgart, Germany

<sup>†</sup>Duke University, Durham, NC, USA

<sup>‡</sup>Advantest Europe GmbH

<sup>§</sup>Politecnico di Torino, Italy

**Abstract**—Recent breakthroughs in machine learning (ML) technology are shifting the boundaries of what is technologically possible in several areas of Computer Science and Engineering. This paper discusses ML in the context of test-related activities, including fault diagnosis, post-silicon validation and yield optimization. ML is by now an established scientific discipline, and a large number of successful ML techniques have been developed over the years. This paper focuses on how to adapt ML approaches that were originally developed with other applications in mind to test-related problems. We consider two specific applications of learning in more depth: delay fault diagnosis in three-dimensional integrated circuits and tuning performed during post-silicon validation. Moreover, we examine the emerging concept of brain-inspired hyperdimensional computing (HDC) and its potential for addressing test and reliability questions. Finally, we show how to integrate ML into actual industrial test and yield-optimization flows.

## I. INTRODUCTION

Machine learning (ML) is having a lasting impact on virtually all fields of today's science and technology. As observed by Gaines [1], ML is a "recursive technology": it supports other technologies, such as devices, computers and compilers, that in turn support ML, thus leading to positive feedback loops and exponential growths in capabilities. Design, manufacturing and test of integrated circuits form one of the most sophisticated value-chains in existence. This paper considers the question of how ML techniques can provide benefits specifically to test-related activities within this value-chain.

It is well-known that testing is not restricted to pass-fail outgoing quality assurance. Among test-related activities are diagnosis, yield optimization and post-silicon validation. Diagnosis aims at inferring the root cause of a circuit's failure from its erroneous responses. Diagnostic information aggregated over large populations of failed circuits, together with technology data, enable systematic identification of yield detractors and thus yield optimization. Post-silicon validation refers to producing a limited series of physical silicon before the start of high-volume manufacturing and taking physical measurements on such "first-silicon" circuits. These measurements give valuable information about the general functionality of the circuit, the attainable performance and power consumption, and the (parametric) yield to be expected in high-volume manufacturing.

All mentioned test-related tasks can profit from ML, and this paper elaborates on that. ML has been applied in recent years to a number of important test and diagnosis problems, ranging from adaptive testing [2–4], optimization for design-for-test [5], yield analysis and characterization [6–8], and fault diagnosis [9–11]. In particular, ML is now also being used to enable the maturation pathway for emerging technologies [12].

In this paper, we will first discuss two specific applications of ML to test-related problems: fault diagnosis in three-dimensional integrated circuits in Section II and tuning during post-silicon validation in Section III. In Section IV, the focus is on a particular ML technique: brain-inspired hyperdimensional computing (HDC). We apply this technique to two representative problems: predicting transistor degradation within SRAM cells and identification of systematic manufacturing process issues from wafer maps. Section V provides an industrial perspective on ML in yield optimization and test, outlining its potentials but also discussing current obstacles and suggesting how to overcome them.

## II. ML-ENABLED DELAY FAULT DIAGNOSIS IN 3D ICs

In this section we describe how ML, in particular Graph Neural Networks (GNNs), have been used to facilitate fault diagnosis for monolithic three-dimensional (3D) integration [12]. As Moore's law reaches physical limits, 3D integration is now being adopted for integrated circuits (ICs). In particular, monolithic 3D (M3D) integration has emerged as a promising technology to achieve higher performance and lower power consumption compared to 2D and die/wafer bonded 3D ICs. M3D leverages fine-grained monolithic inter-tier vias (MIVs) to achieve high-precision alignment and extremely thin device layers. The size of MIVs is of the same order of magnitude as conventional back-end-of-line (BEOL) vias. As a result, a large number of MIVs can be used in M3D designs, leading to a significant reduction in wirelength.

Despite these advantages, M3D introduces several challenges. Fabricating upper-tier transistors in M3D designs with typical thermal budgets causes damage to wires and cells underneath. While advanced processes have been developed to fabricate transistors at a low temperature, they can cause up to 20%

performance mismatch between the devices in different tiers. The reliability of interconnects is another concern for M3D ICs. Standard copper/low-k BEOL cannot be used between tiers because the fabrication steps in the upper tiers pose contamination risks, while low-k dielectrics are thermally unstable after annealing processes [7]. Moreover, MIVs in M3D designs are prone to defects as they penetrate through the inter-tier dielectric. Surface roughness can produce voids in the dielectric [8], which may lead to voids in MIVs during etching, resulting in delay defects and degradation of circuit performance. Delay-fault diagnosis is therefore important in order to provide early feedback to the foundry and facilitate yield learning.

In contrast to die/wafer bonding in stacked 3D integration, tiers in M3D designs are fabricated in situ, which makes it hard to ascertain a known-good tier before assembly. Delay-fault diagnosis catered to M3D designs is especially important as existing diagnosis methodologies cannot provide the high level of resolution (i.e., fault localization) needed at the tier level. To make M3D integration feasible, there is a need for a diagnosis framework that can efficiently localize faults to a tier. Such a diagnosis framework should provide early feedback to the foundry before the time-consuming physical failure analysis. An effective diagnosis method should also be compatible with existing diagnosis flows provided by commercial tools to improve the quality of diagnosis.

The framework recently proposed in [12] aims at using graph neural networks (GNNs) to improve diagnostic resolution for M3D designs. Tier-level predictions are used to enhance the quality of diagnosis reports generated by an automatic test pattern generation (ATPG) tool. This is a key benefit of the proposed solution; it is synergistic and compatible with commercial tools. In addition, ML-aided MIV diagnosis can help in the early characterization of defective MIVs.

GNN is an ML method that processes data on graphs. In the field of IC design, GNN has gained special attention because it can carry out computations directly in non-Euclidean domains. ML models such as recurrent neural networks and convolutional neural networks are not effective for graph-structured data because they operate on Euclidean data such as images and text sequences. However, different graphs have different numbers of nodes/edges and irregular node connections. A preprocessing phase is therefore required to map graph structures to simplified representations, while topological dependency of each node may be lost during this phase. For diagnosis problems, GNN models can learn the complex, non-linear relationship between a fault location (root-cause) and the failure response.

In [12], a diagnosis solution for M3D based on GNN has been presented to locate faults at the tier level. At-speed transition delay fault diagnosis was considered because the M3D-specific defects tend to be manifested in the form of delay faults that impact circuit timing. The proposed method is able to localize faults based on the circuit netlist and failure log files from the tester. In particular, two models, namely Tier-predictor and MIV-pinpointer, based on GNNs have been developed to locate

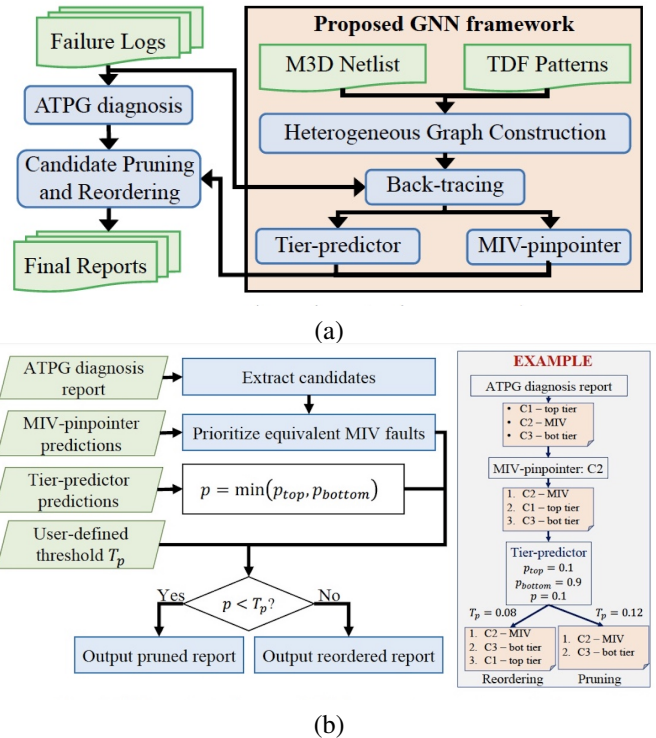


Fig. 1. Overview of diagnosis flow from [12] (a) and details on candidate pruning and reordering (b).

faults at the tier level and in MIVs, respectively. The prediction results provide quick feedback to the foundry or diagnosis team prior to time-consuming failure analyses. This approach also provides a candidate reordering and pruning algorithm based on these predictions to improve the quality of ATPG diagnosis reports. The results show that with less than 1% loss of accuracy, diagnostic resolution is significantly improved for the OpenCore and ISPD benchmarks. Fig. 1 illustrates the diagnosis flow using this approach.

### III. LEARNING TO TUNE IN POST-SILICON VALIDATION

Key concerns in post-silicon validation (PSV) are to ensure functional correctness and to guarantee to stay within given performance limits. To this end, a first set of devices is produced of which physical measurements are obtained. Ideally, this set of devices is representative for the subsequent high-volume manufacturing. As such, it contains devices with differences in performance and even outliers (devices with faulty behavior in certain regions, i.e., for specific parameter combinations).

Robustness against process variations, degradation effects or unintended side effects due to non-optimal design implementations is ensured by tuning. So-called tuning knobs, which are configurable registers on the devices, are set to satisfy above requirements under a range of operating conditions. This includes adjusting bias settings, clock frequencies, voltages or currents. In static settings, the tuning knobs are chosen optimally with respect to a given figure of merit (e.g., power

consumption or temperature). In a dynamic setting, they can be adjusted depending on operating conditions such as temperature or certain operation modes. This allows not only an optimal tuning of the devices immediately after manufacturing but even to cope with degradation of devices over time while they are deployed in field.

A figure of merit  $f(\vec{x}, \vec{c}, \vec{t})$  that we aim to optimize (here: maximize) depends on the tuning knobs  $\vec{t}$ , the conditions  $\vec{c}$  and metadata  $\vec{x}$  (the classical stimuli for testing). All parameters can be real, integer or nominal/categorical values. The latter include different operation modes. A high number of parameters, however, leads to the so-called curse of dimensionality [13]: Any exhaustive search of their parameter space is infeasible due to the combinatorial explosion of their values' combinations.

Conventionally, PSV offers no systematic solutions to the tuning problem, thus it requires experts to judge where to sample in the parameter space, and which directions to explore. However, rising complexity of modern ICs and tight integration in the sub 10 nm process range increases the difficulties to solve this intricate optimization problem. With unexpected side-effects and an increase in the number of affordable tuning knobs, expert knowledge can even become misleading.

Assuming that we treat the devices under test as black-boxes and that we do not assume any knowledge about the design process, an uninformed (random) sampling of the parameter space leads to a single data set for each of the  $N$  devices. Based on the overall data, we can learn and then optimize the device performance function  $f$ , solving a complicated, bound-constrained, mixed-type optimization task. The scenario resembles hyperparameter tuning for deep neural networks [14]. In PSV, however, point-wise methods such as Random Search or Grid Search are too inefficient to be used in practice, and gradient-based approaches have difficulties to cope with mixed-type settings that include non-metric (nominal) parameters. Furthermore, we are interested in tuning that is robust against faulty devices (outliers). The aforementioned problems get worse with the number of parameters; thus whenever possible, the dimensionality of the search space can be reduced to the set of most relevant parameters [15] in a preprocessing step. Nonetheless, new approaches are required to intelligently learn from data for efficient and robust performance tuning in PSV.

We have recently proposed a two-stage process that ensures robustness in the tuning process [16]. In a first step, a NN learns  $f_i(\vec{x}, \vec{c}, \vec{t})$  separately for each device. Note that while we obtain an ensemble of trained predictors, this differs from ensemble learning as each predictor is trained on different data.

We then learn a soft-min combination rule based on neural networks trained for each device to obtain an approximation  $\hat{f}(\vec{c}, \vec{t}) = \text{soft-min}\{f_i, i = 1..m\}$  that represents the robust worst case for all  $\vec{x}$ .

In a second step, and to meet fast response times for tuning, we have introduced learn-to-optimize [17] to PSV and leverage reinforcement learning (RL) methods as in [18–20] to learn an optimal response  $\vec{t}^*$  for each set of conditions  $\vec{c}$ , see Fig. 2. A

robust tuning law

$$f(\vec{c}) = \arg \max_{\vec{t}} \hat{f}(\vec{c}, \vec{t})$$

is then obtained.

An appealing property of the learn-to-optimize approach is its fast time-to-optimize. As soon as the tuning law has been learned, the optimization for each  $\vec{c}$  is fast. Compared to the Tree-structured Parzen Estimator (TPE) algorithm [14, 21] and the Powell method [22] as baselines, we have been able to improve by a factor of about 100 in time. And we even outperform an approximate version (similar computing budget to learned tuning law) of Powell's method by a factor of about 35; see [17] for further information.

In summary, PSV provides new and special challenges to data-driven methods. The robust learn-to-optimize approach has appealing properties that match the characteristics and demands of tuning in PSV very well.

#### IV. BRAIN-INSPIRED MACHINE LEARNING FOR SEMICONDUCTOR TEST AND RELIABILITY

Common machine-learning methods require large datasets to train on in order to identify and “learn” the underlying patterns. However, each (prototype) sample is valuable in the area of semiconductor manufacturing. In deep neural networks (DNNs), their learning capability is based on neurons and in turn on expensive floating-point matrix multiplications. Hence, DNNs require a lot of processing power and time for training and inference, preventing an embedding into the test equipment and thus decisions at the edge. Lastly, those common methods struggle with noise in measurements, e.g., only a few pixel difference in an image can cause a wrong classification.

In this section, we describe how the emerging concept of brain-inspired hyperdimensional computing (HDC) does address the constraints and challenges in the field of semiconductor test. The concept of HDC is based on large vectors, hypervectors, to represent real-world data and complex patterns [23]. By abstracting the data into hyperspace, small changes in the patterns and noise can be compensated. HDC has been employed for gesture recognition [24], seizure detection [25], language classification [26], and others.

The typical dimension of such hypervectors is 10 000. It can consist of simple bits, bipolar values, integers or real numbers. The initial simple value-representing hypervectors are generated randomly. Due to their high dimension, errors like bit flips in single components do not impact the overall hypervector in a meaningful way making it very robust against noise. The similarly for two binary hypervectors, consisting only of ones and zeros, is computed with the Hamming distance. To create complex pattern-representing hypervectors, the real-world data is encoded into the hyperspace.

In the first scenario, we describe how HDC can be employed to infer the degradation of the transistors inside an SRAM cell and other circuits. Issues in manufacturing processes or runtime degradation (aging) can be detected that impact all

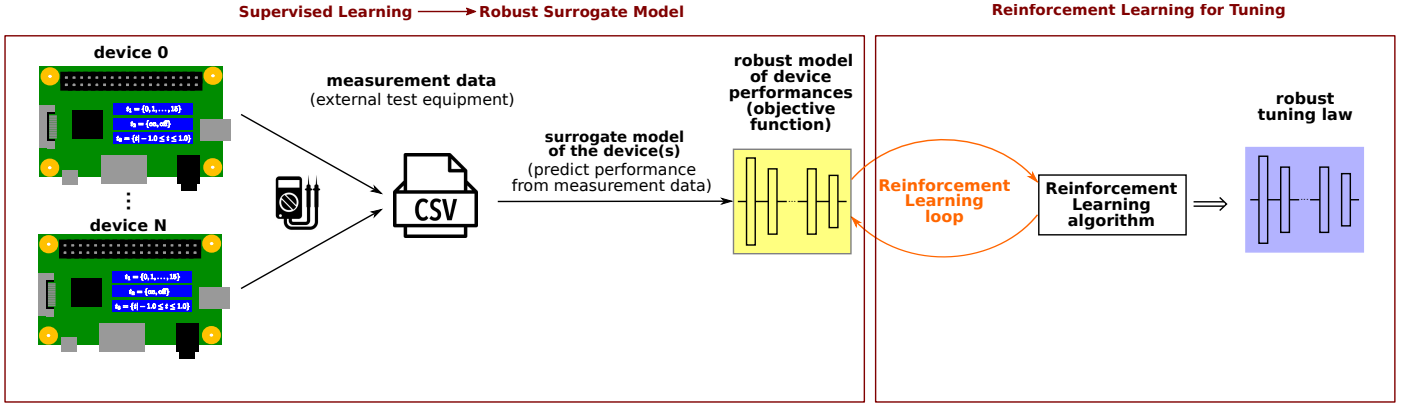


Fig. 2. Overview of the robust learn-to-optimize approach to tuning in post-silicon validation, adapted from [17].

the underlying transistors. To encode a waveform, e.g., the signal response of a device under test to an input voltage, it is first quantified in the time and value domain. For each of the value levels, a random hypervector is generated. The first  $n$  values are mapped into hyperspace by using the corresponding hypervectors. The  $i$ -th hypervector in this  $n$ -gram is permuted (i.e., circular shift or rotated)  $i$  times to encode the temporal dependency. All  $n$  permuted hypervectors are bundled into one by a component-wise majority vote. If there are more ones than zeros at a given position in the input hypervectors, the resulting hypervector has a one, otherwise a zero. Hence, the dimension does not change. The component-wise nature enables a high degree of parallelization, the simple comparisons are light-weight computations. The above steps are repeated with the remaining values in the waveform. Those computed hypervectors are again bundled by applying the same component-wise majority vote resulting in a single hypervectors representing the whole signal response of the DUT. Through this encoding, noise in the signal can be compensated as the individual  $n$ -gram hypervectors contribute little to the resulting hypervector.

Such a response could be the butterfly curve of an SRAM cell if the input voltage is swept. From this curve, the shift in threshold voltage  $\Delta V_{th}$  can be derived. Such a shift can be due to manufacturing defects, process variation, aging, or temperature change. However, the  $\Delta V_{th}$  cannot be easily measured directly. Instead, we propose to train a model with simulation data and infer the  $\Delta V_{th}$  from the measured signal response. Those  $\Delta V_{th}$  values are associated with the butterfly curves generated from the simulated voltage sweep from 0.0 V to 0.7 V. The  $\Delta V_{th}$  values are set in steps of 10 mV from 0 mV to 100 mV for nFinFET and pFinFET. If we want to detect systematic manufacturing defects, then a  $\Delta V_{th}$  only applies to pFinFET or nFinFET at the same time.

With out brain-inspired HDC model, we can infer the  $\Delta V_{th}$  with an average accuracy of 3.7 mV. To compensate process variation, we measure 64 different SRAM cells of the same chip. Each has a different  $\Delta V_{th}$  due to the variation, but all share the underlying defect or aging-driven  $\Delta V_{th}$ . If their mean  $\Delta V_{th}$  confirms with the simulations, the chip is defect free

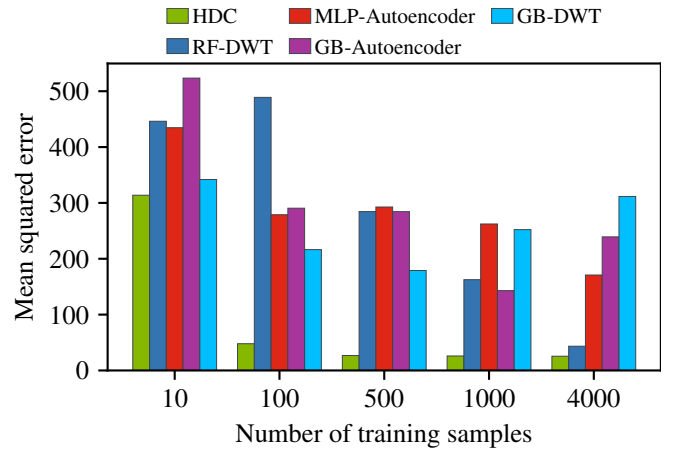


Fig. 3. HDC requires significantly less training samples than other classifiers like random forest (RF), multilayer perceptron (MLP), or gradient boosting (GB) combined with the feature extractors discrete wavelet transform (DWT) and autoencoder.

and the measurements can be used to estimate the impact of process variation. Compared to other methods, HDC requires significantly less training samples to achieve better accuracies as shown in Fig. 3.

Another challenge is the identification of systematic issues in the manufacturing process. One early indicator are wafer-level tests, which classify chips of a wafer as functional or broken. The result is presented as an image, the wafer map, in which certain patterns can be identified, e.g., a doughnut or a cluster of broken chips in the center. To detect such patterns automatically, various ML methods have been employed, including CNNs. With HDC, the positions of all broken chip is bundled into a single hypervector. To also capture rotations and variations, multiple hypervectors can represent the same defect pattern. Additionally, the computed Hamming distances are used by a simple NN for classification, turning HDC into a feature extractor. With this approach, we achieve accuracies of about 95% on the WM-811K dataset, which is at the same level as other methods. However, the training 13x faster than SVM and inference time is 42x faster than a CNN [27].

## V. INDUSTRIAL SOLUTIONS FOR MACHINE-LEARNING ENABLED YIELD OPTIMIZATION AND TEST

Applying machine learning and derived techniques in the area of semiconductor manufacturing and test has exhibited significant growth and attention recently. As in many other industries, data science approaches that combine learning and algorithmic methodologies together with large amounts of data have demonstrated their specific advantages. In particular, approaches driven by data analytics have been widely employed to improve yield (e.g., by harvesting borderline devices) and to improve test cost (e.g., by identifying redundant tests).

However, these methodologies are typically evaluated in a restricted environment. Especially, research institutions are constrained by limited data available and often have to rely on obfuscated, sometimes even synthetic, data sets. Hence, experiences on applying such approaches in real-world scenarios using actual data are very seldom. Even more seldom is access to the required infrastructure and techniques to apply ML-based techniques in the complicated production environments of today's semiconductor value chains and evaluate their effectiveness.

In the following, we will point towards some particular specialties that need to be solved to transition a given methodology to a real-world use case.

### A. Representative Data Sets

Due to the unavoidable influences of process variation, the intrinsic characteristics of semiconductors are constantly changing over time. Especially between lots, but for modern processes also between wafers and even dies, the electrical and spacial properties of an individual chip are different. For learning-based techniques, this is a particular challenge as they learn the characteristics of the training data and apply them during the inference. Hence, in case of underlying drifts in the processes, the quality of the model is subject to degradation.

To counter this effect, one approach is to learn on a comprehensive initial training set that covers the full (expected) variation space. Hence, robustness against variations in increased at the cost of increased data quality requirements.

### B. Data Availability Across the Semiconductor Value Chain

As mentioned, one of the most fundamental requirements to train data analytics techniques is the availability of data meeting the requirements for the data analysis at hand.

In order to meet these requirements, particular efforts need to be spent during design of the device but also when defining the supply chain.

As a particular example, die-level traceability needs to be established by electronically or optically readable device identifiers when relating measurements from one insertion to the next. Due to the importance for analytics, die-level traceability is getting a de-facto standard for reasonably sized digital devices.

Even more trivially conceptually, the required data needs to be available for analysis, which implies the need to establish proper

communication channels. In an integrated device manufacturer (IDM)-like setup, where the design, manufacturing as well as data analysis teams are part of the same company, silos between the teams hinder efficient information flows and hence need to be avoided.

However, the same scenario is much more complex in a generalized case for an outsourced environment requiring the fabless design house to fully establish data transfer from the worldwide distributed outsourced semiconductor assembly and test organizations (OSAT)s using the measurement equipment from the employed testers. In such a case, appropriate data privacy levels need to be established between all involved partners using a combination of technical and contractual means.

### C. Frequent Changes to the Model

Production test floors are traditionally very adverse to any changes in the production process, especially in safety-critical areas, such as automotive chips. This is driven by the work required to do qualify of a given production and test setup including its hardware as well as software components.

However, this classical paradigm is challenged when models need to be updated frequently to maintain their accuracy. From a methodology perspective, a model update requires a re-qualification to ensure the desired behavior and reduce the risk of unwanted side effects. Accordingly, also the qualification methodologies need to evolve to be driven by algorithms and acceptance criteria to monitor.

### D. Infrastructure Needs

Summarizing the above challenges, traditional realities of the semiconductor value chain are facing a constant pressure to embrace change to support machine-learning based techniques and, more generally, *Smart Manufacturing*.

As a result, technology trends originating from more general software development processes are gaining momentum in the semiconductor test as well.

- Containerized software-packages allow a greater flexibility to deploy and run algorithms on a multitude of environments, including cloud, on-premise as well as during the test.
- Dedicated companion environments integrated into test floors allow execution of workload intensive computation workloads while protecting the data and algorithms.
- Continuous deployment techniques allow frequent changes updating while constantly monitoring the correctness and validity of the environment
- Real-time data streams provide connectivity between the measurement equipment (e.g. the tester at an OSAT) drive fast feedback loops with low latency.
- Big data techniques such as no-SQL data storages enable efficient data analytics techniques on large data volumes.

Still, orchestrating these technologies together is a huge investment into the necessary software and infrastructures that requires very specific knowledge and detailed technical alignment between the involved partners and production locations.

Hence, open infrastructures connecting the various test floor level components with their counterparts on the cloud or on-premise data to allow data and control flow in are an attractive alternative to a make strategy.

## VI. CONCLUSION

The efficiency of test, diagnosis, post-silicon validation and yield optimization can significantly profit from state-of-the-art machine learning techniques. In this paper, we elucidated this new development from several directions. We provided two specific examples where suitable ML techniques were successfully applied to test-related problems. We elaborated on the potential of brain-inspired hyperdimensional computing for further test-related questions. Finally, we considered the integration of ML into test flows from an industrial perspective. We believe that more interdisciplinary research, connecting specialists in test and in ML, will pave the way for further advances in this area.

## ACKNOWLEDGEMENT

The work of H. Amrouch, D. Pflüger and I. Polian was supported by Advantest as part of the Graduate School “Intelligent Methods for Test and Reliability” (GS-IMTR) at the University of Stuttgart. The work of K. Chakrabarty was supported in part by the National Science Foundation under grant CCF-1908045.

## REFERENCES

- [1] B. R. Gaines, “A conceptual framework for stochastic neuromorphic computing,” *IEEE Design & Test*, vol. 38, no. 6, pp. 16–27, 2021.
- [2] M. Liu, R. Pan, F. Ye, X. Li, K. Chakrabarty, and X. Gu, “Fine-grained adaptive testing based on quality prediction,” *ACM Trans. Design Autom. Electr. Syst.*, vol. 25, no. 5, pp. 38:1–38:25, 2020.
- [3] M. Chen and A. Orailoglu, “Test cost minimization through adaptive test development,” in *26th International Conference on Computer Design, ICCD 2008, 12-15 October 2008, Lake Tahoe, CA, USA, Proceedings*, IEEE Computer Society, 2008, pp. 234–239.
- [4] H. D. Stratigopoulos and C. Streitwieser, “Adaptive test flow for mixed-signal ics,” in *35th IEEE VLSI Test Symposium, VTS 2017, Las Vegas, NV, USA, April 9-12, 2017*, IEEE Computer Society, 2017, pp. 1–6.
- [5] Z. Li, J. E. Colburn, V. Pagalone, K. Narayanun, and K. Chakrabarty, “Test-cost optimization in a scan-compression architecture using support-vector regression,” in *35th IEEE VLSI Test Symposium, VTS 2017, Las Vegas, NV, USA, April 9-12, 2017*, IEEE Computer Society, 2017, pp. 1–6.
- [6] N. Sumikawa, D. G. Drmanac, L. Wang, L. Winemberg, and M. S. Abadir, “Forward prediction based on wafer sort data - A case study,” in *2011 IEEE International Test Conference, ITC 2011, Anaheim, CA, USA, September 20-22, 2011*, IEEE Computer Society, 2011, pp. 1–10.
- [7] N. Sumikawa, J. Tikkanen, L. Wang, L. Winemberg, and M. S. Abadir, “Screening customer returns with multivariate test analysis,” in *2012 IEEE International Test Conference, ITC 2012, Anaheim, CA, USA, November 5-8, 2012*, IEEE Computer Society, 2012, pp. 1–10.
- [8] N. Sumikawa, L. Wang, and M. S. Abadir, “An experiment of burn-in time reduction based on parametric test analysis,” in *2012 IEEE International Test Conference, ITC 2012, Anaheim, CA, USA, November 5-8, 2012*, IEEE Computer Society, 2012, pp. 1–10.
- [9] M. Liu, F. Ye, X. Li, K. Chakrabarty, and X. Gu, “Board-level functional fault identification using streaming data,” *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.*, vol. 40, no. 9, pp. 1920–1933, 2021.
- [10] M. Liu, X. Li, K. Chakrabarty, and X. Gu, “Knowledge transfer in board-level functional fault diagnosis enabled by domain adaptation,” *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.*, vol. 41, no. 3, pp. 762–775, 2022.
- [11] Y. Tian et al., “A supervised machine learning application in volume diagnosis,” in *24th IEEE European Test Symposium, ETS 2019, Baden-Baden, Germany, May 27-31, 2019*, IEEE, 2019, pp. 1–6.
- [12] S.-C. Hung, S. Banerjee, A. Chaudhuri, K. Chakrabarty, and K. Chakrabarty, “Graph neural network-based delay-fault localization for monolithic 3D ICs,” in *Design Automation and Test in Europe Conf. (DATE)*, 2022.
- [13] R. Bellman, *Adaptive Control Processes: A Guided Tour*, ser. Rand Corporation. Research studies. Princeton University Press, 1961.
- [14] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, “Algorithms for hyper-parameter optimization,” *Advances in neural information processing systems*, vol. 24, 2011.
- [15] Y. Liao, R. Latty, and B. Yang, *Feature selection using batch-wise attenuation and feature mask normalization*, 2020. arXiv: 2010.13631.
- [16] P. Domanski, D. Pflüger, R. Latty, and J. Rivoir, “ORSA: Outlier robust stacked aggregation for best- and worst-case approximations of ensemble systems,” in *2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2021, pp. 1357–1364.
- [17] P. Domanski, D. Pflüger, J. Rivoir, and R. Latty, *Self-learning tuning for post-silicon validation*, 2021. arXiv: 2111.08995.
- [18] K. Li and J. Malik, “Learning to optimize,” *arXiv preprint arXiv:1606.01885*, 2016.
- [19] O. Wichrowska et al., “Learned optimizers that scale and generalize,” in *International Conference on Machine Learning*, PMLR, 2017, pp. 3751–3760.
- [20] Y. Chen et al., “Learning to learn without gradient descent by gradient descent,” in *International Conference on Machine Learning*, PMLR, 2017, pp. 748–756.
- [21] J. Bergstra, D. Yamins, and D. Cox, “Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures,” in *International conference on machine learning*, PMLR, 2013, pp. 115–123.
- [22] M. J. Powell, “An efficient method for finding the minimum of a function of several variables without calculating derivatives,” *The computer journal*, vol. 7, no. 2, pp. 155–162, 1964.
- [23] P. Kanerva, “Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors,” *Cognitive Computation*, vol. 1, pp. 139–159, 2009.
- [24] A. Moin et al., “A wearable biosensing system with in-sensor adaptive machine learning for hand gesture recognition,” *Nature Electronics*, vol. 4, no. 1, pp. 54–63, 2021.
- [25] A. Burrello, K. Schindler, L. Benini, and A. Rahimi, “One-shot learning for iieg seizure detection using end-to-end binary operations: Local binary patterns with hyperdimensional computing,” in *2018 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, 2018, pp. 1–4.
- [26] G. Karunaratne, M. Le Gallo, G. Cherubini, L. Benini, A. Rahimi, and A. Sebastian, “In-memory hyperdimensional computing,” *Nature Electronics*, pp. 1–11, 2020.
- [27] P. R. Genssler and H. Amrouch, “Brain-inspired computing for wafer map defect pattern classification,” in *IEEE Int’l Test Conf. (ITC’21)*, 2021.