On-the-fly Defect-Aware Design of Circuits based on Silicon Dangling Bond Logic

Jan Drewniok, Marcel Walter, Samuel Sze Hang Ng, Konrad Walus, and Robert Wille
https://www.cda.cit.tum.de/research/nanotech/

***Abstract*— *Silicon Dangling Bonds* (SiDBs) have emerged as a promising post-CMOS technology for achieving ultra-low power dissipation, establishing themselves as a highly anticipated and environmentally friendly competitor in the realm beyond conventional CMOS. To support the SiDB logic framework, design automation approaches have rapidly evolved. However, at the atomic scale of SiDBs, material imperfections pose a significant roadblock in scaling these devices. Consequently, established design automation flows, which are defect-agnostic, are inadequate and have not kept pace with the latest experimental findings and advances in fabrication capabilities. A first attempt was recently proposed that extends established defect-agnostic physical design methods by rudimentary defect-aware capabilities. While promising at first glance, in this work, we show that this first attempt yields unsatisfactory results. Subsequently, we present a novel approach that automatically designs a tailored SiDB gate on-the-fly whenever an SiDB gate encounters atomic defects in its vicinity, thereby incorporating these atomic defects into its layout as an integral part. Our experimental evaluations confirm that the proposed approach is capable of designing SiDB circuits of significant complexity and size, even in the presence of atomic defects for the first time. Therefore, this work contributes to advancing this promising post-CMOS technology.**

I. Introduction

The 2023 Nobel Prize in Chemistry for the discovery and synthesis of quantum dots confirms the profound importance and potential of nanotechnology [1], [2]. While these quantum dots with up to several thousand atoms have been studied for decades, *Silicon Dangling Bond* (SiDB) quantum dots have recently entered the scene and have already established themselves as promising candidates for the beyond-CMOS era of computing [3]–[6]. SiDBs can be manufactured at atomically precise locations on the spatially periodic H-Si(100)-2×1 surface, employing the tip of a *Scanning Tunneling Microscope* (STM, [4]). Importantly, they possess the ability to maintain discrete charge states (encompassing negative, neutral, and positive), which change conditionally in the vicinity of other SiDBs due to electrostatic field coupling. These properties can be utilized to construct logic circuitry.

In fact, the successful fabrication of wires and gates has already been demonstrated [3], [4], [7]. This revolutionary logic platform has garnered immense interest. As a result, the design automation capabilities have evolved rapidly to support the SiDB technology framework [8]–[12]. Furthermore, several highly-efficient physical simulators, including *QuickSim* [13], *QuickExact* [14], and *SimAnneal* [15], have been introduced. The availability of these tools accelerates the exploration of novel SiDB logic designs and their applications.

However, a significant barrier to revealing the full potential of SiDB logic is the presence of atomic defects, particularly those located on or near the surface, whose occurrences are unavoidable with contemporary fabrication capabilities [16]–[18]. Atomic defects include a range of structural and chemical irregularities within the silicon crystal lattice that affect the SiDB logic. Consequently, established design automation flows, which are defect-agnostic, are inadequate.

A first attempt was recently proposed that extends established defect-agnostic physical design methods by rudimentary defect-aware capabilities. While promising at first glance, in this work, we show that this first attempt yields unsatisfactory results. Thus, there is no sufficient design automation solution to design SiDB circuits in the presence of atomic defects. This urgently calls for a more sophisticated and radical approach.

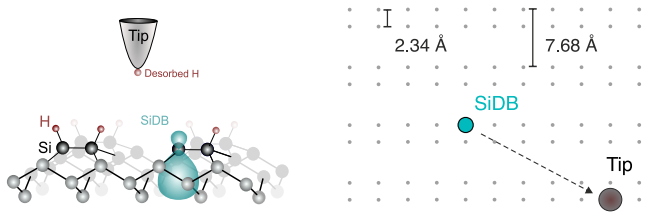
In light of this discovery, this work then introduces an alternative method: in the case of a conflict between an SiDB gate and atomic defects, a tailored SiDB gate is designed on-the-fly—integrating these atomic defects as an integral part of its design. This eliminates the need to strategically place and route SiDB gates while avoiding atomic defects, ultimately allowing the design of SiDB circuits of significant size and complexity in the presence of atomic defects.

The main results show, first, that designing SiDB circuits on surfaces with substantial defects is indeed not feasible by merely avoiding atomic defects. Second, it is shown that the proposed approach is capable of designing SiDB circuits of relevant size and complexity on significantly defective surfaces.

The remainder of this work is structured as follows: Section II reviews preliminaries—the SiDB logic platform and physical design. Subsequently, Section III serves to motivate the need for a novel physical design approach for defective surfaces. By showing that established physical design approaches and a recently proposed attempt fail to design SiDB circuits of relevant size and complexity on significantly defective surfaces, it is emphasized that the problem to be solved is non-trivial and that it is important to develop a novel approach to keep pace with the latest experimental findings in fabrication. The general idea of our more sophisticated

Jan Drewniok, Marcel Walter, and Robert Wille are with the Chair for Design Automation, Technical University of Munich, Germany. Marcel Walter is also with the University of Bremen, Germany. Robert Wille is also with the Software Competence Center Hagenberg GmbH (SCCH), Austria. E-mail: {jan.drewniok, marcel.walter, robert.wille}@tum.de

Samuel Sze Hang Ng, and Konrad Walus are with the Department of Electrical and Computer Engineering, University of British Columbia, Canada. Email: {samueln, konradw}@ece.ubc.ca



(a) SiDB fabrication on the H-Si(100)-2×1 surface (side view). (b) SiDB fabrication on the H-Si(100)-2×1 surface (top view).

Fig. 1: The SiDB logic platform.

and radical approach is presented in the same section. The details are explained in Section IV. In order to establish the challenging nature of physical design on defective surfaces, and at the same time to demonstrate the applicability of our proposed method to successfully design SiDB circuits, an experimental study on realistic surface data is performed in Section V. Finally, Section VI summarizes and concludes this work.

II. Preliminaries

This section reviews the necessary preliminaries for the comprehension of the remainder of this manuscript. First, an overview of the SiDB logic platform is given in Section II-A, followed by a review of its physical design in Section II-B.

A. The SiDB Logic Platform

SiDBs are created on an H-Si(100)-2×1 surface at the single-atom scale using an atomically sharp tip of a *Scanning Tunneling Microscope* (STM) [4], [5], [18]–[21]. An applied voltage breaks the covalent bond between hydrogen and a silicon atom—allowing the hydrogen atom to be desorbed to the tip, leaving behind an open valence bond (sp^3-orbital) called SiDB.

Example 1: Figure 1a and Figure 1b schematically sketch the SiDB fabrication process on an H-Si(100)-2×1 surface.

Any SiDB can be occupied by up to two electrons [22]. However, the number of electrons within the SiDB and, thus, its charge state, depends on the local electrostatic potential, which can be influenced by external electrodes or neighboring SiDBs. This property of exhibiting multiple charge states while being isolated from the conduction band is essential for the construction of logic gates and devices [3], [4], [22].

This pioneering research has led to the successful fabrication of wires consisting of *Binary-Dot Logic* (BDL) pairs [3], [4]. BDL pairs typically consist of two SiDBs sharing an electron. Depending on where the electron is predominantly located, the pair encodes either a `1` or a `0`. In addition, SiDB gates with a footprint smaller than $30\,\mathrm{nm}^2$ have already been produced [3].

B. Physical Design of SiDBs

Recent achievements in the fabrication capabilities and commercialization efforts in the area of SiDB logic have sparked a growing interest in design automation techniques for this technology [23]–[25]. This has led to the development of initial computer-aided design tools, physical simulators, and physical design algorithms [8], [9], [13]–[15], [26].

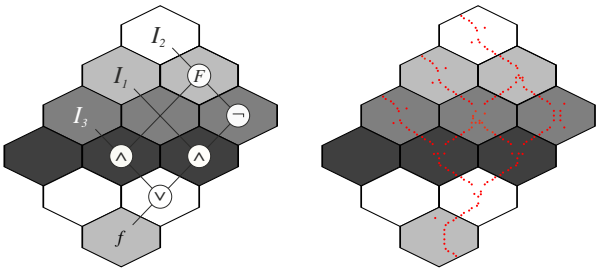
Physical design involves implementing a dot-accurate layout by assembling elementary gates on a surface and connecting them with wire segments. The goal is to create a circuit that retains the functionality specified by a gate-level netlist, usually obtained from a preceding logic synthesis and technology mapping process. However, in the SiDB technology, a circuit has to be divided into uniform regions that can be alternately activated and deactivated by external fields to stabilize and regulate the direction of the signal [8]. An activated region maintains its logic states and performs computations, while deactivated regions act as barriers to reduce unwanted signal interference. This process is known as *electrostatic clocking* or *SiDB clocking* [15].

SiDB clocking is expected to be achieved by manipulating the surface charge states by external electrodes. This means that segments of the system can be disabled by eliminating surface charges—resulting in the creation of electrically neutral regions [15]. Therefore, the SiDB surface is divided into regions, called *tiles*. Each tile can implement a Boolean function. The dot-accurate layout can then be obtained either by using a standard SiDB gate library or by designing SiDB gates on-the-fly. Thus, the physical design process starting from a netlist and ending with a dot-accurate layout consists of two coarse main steps (which are usually broken down into individual sub-steps): 1) Obtaining a gate-level layout representation for the constraints given by the clocked tiles (proven to be $\mathcal{NP}$-complete [27]), and, 2) Mapping each gate tile to a dot-accurate SiDB implementation.

The initial physical design process for the SiDB technology was introduced by Walter *et al.* [8]. In that study, the authors present a solution for each sub-step of the physical design process for SiDB logic—starting with a gate-level netlist, proceeding to the creation of a gate-level layout via exact physical design and taking clocking constraints into account, and, finally, using a dot-accurate gate library, the *Bestagon* gate library, to end with the final circuit [8].

Example 2: Figure 2 illustrates the physical design process of a simple 2:1-multiplexer implemented in the SiDB technology. The process begins with a gate-level netlist (obtained from logic synthesis), specifying the multiplexer's functionality. In the netlist, function labels are assigned to each node, and, thereby it is similar to a combinational circuit, but it disregards physical characteristics and only takes into account the logic level. This netlist is to be mapped onto the clocked and tiled surface. A placement and routing algorithm determines such a mapping, illustrated in Figure 2a. Finally, each placed gate and wire segment is converted to an SiDB representation, yielding a dot-accurate layout as shown in Figure 2b. This process is done by applying a pre-determined gate library. In this example, the standard SiDB *Bestagon* gate library is used [8].

The *Bestagon* gate library has been designed and validated by physical simulations. Inspired by the SiDB gates manufactured by Huff *et al.* [4], the *Bestagon* gates also consist of pins (the number is defined by the dimension of the Boolean function that is implemented) and one output BDL wire. However, to allow for a straight-forward mapping from each gate tile to a dot-accurate SiDB implementation and, thereby, ending up with complex circuits, the pins are



(a) Gate-level layout fulfilling the SiDB technology constraints. Different shades of gray indicate different clock numbers.

(b) Dot-accurate layout obtained by applying the *Bestagon* gate library [8]. Each red dot indicates one SiDB.

Fig. 2: Simplified physical design flow.

standardized in the sense that any SiDB implementation of a Boolean function of the same dimension possesses pins at the same position within the tile (neglecting mirroring). Only the different location of the SiDBs in the area, called *canvas*, between the pins, called the *skeleton*, decides which Boolean function the SiDB layout fulfills. Therefore, when designing a new single SiDB gate for a standard library, the goal is to find the correct SiDB arrangement within the canvas so that the correct output is obtained for all input combinations.

III. Motivation & General Idea

In recent years, significant progress has been made in developing a physical design workflow for the SiDB technology as reviewed above. This workflow—starting from an abstract logic level and culminating in a dot-accurate representation verified by efficient physical simulators—has enabled the design of intricate SiDB circuits [8], [9]. However, H-Si(100)-2×1 surfaces are prone to material imperfections at the atomic scale, so-called atomic defects, as recently quantified experimentally [17], [18], [28]. Established SiDB design automation flows, which are defect-agnostic, have not kept pace with the latest experimental findings and advances in fabrication capabilities. In Section III-A, we discuss the resulting challenges—constituting the motivation of this work. Afterward, in Section III-B, we sketch the general idea of a solution that addresses these challenges and is proposed in this work.

A. Challenges in SiDB Physical Design

Despite decades of optimization in silicon crystal growth that massively minimized fabrication imperfections, atomic defects are still prevalent. Croshaw *et al.* [17] were able to characterize 13 different types of atomic defects routinely found on H-Si(100)-2×1 surfaces, which can be divided into two defect categories: *charged* and *neutral*. Since SiDB gates are only comprised of a few SiDBs each, atomic defects on the surface and shallow subsurface regions can significantly affect both device patterning and gate operation. In particular, charged defects can heavily impact SiDB charge distributions and, thus, the logical behavior of gates. As investigated by Ng *et al.* [16], charged atomic defects can disturb gate operation at a distance of more than 10 nm. However, neutral atomic defects are not to be disregarded either, because they reduce the effective area where SiDB gates can be fabricated. This situation imposes harsh constraints on the placement and routing of SiDB gates.

Example 3: Consider Figure 3a which sketches a scenario in which a neutral and a negative atomic defect are present on a single clocking tile within an H-Si(100)-2×1 surface. In this situation, attempting to create an OR gate using the *Bestagon* gate library will fail. The reason for this failure is twofold: 1) a cascade of neutral defects blocks a position where an SiDB is to be fabricated, and 2) the proximity of the negative defect disturbs the gate operation.

The previous example highlights a significant limitation of using standard gate libraries in the presence of atomic defects: when a standard gate is blocked by defects, the only available option with current physical design approaches is to avoid that surface location entirely, reducing the available surface area.

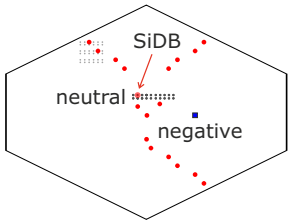
A first attempt recently proposed to apply these established defect-agnostic physical design methods and tries to avoid atomic defects during the design process [26]. This works as follows: first, the SiDB surface is divided into hexagonal tiles. Then, for each gate, the effect of the atomic defects on that gate at a given tile is examined. If the gate cannot be executed due to the atomic defects, the corresponding tile is avoided. This yields a blacklist of gates on certain tiles that would result in fabrication defects or gate failures. Finally, this blacklist is then used as a set of additional constraints in the placement and routing phase of the design process. This effort results in layouts that avoid surface defects.

B. General Idea

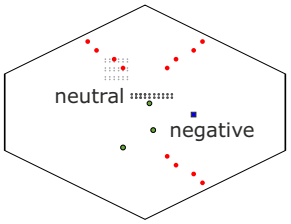
The challenges reviewed above constitute a serious roadblock in the realization of practically-relevant SiDB designs. While the initial attempt to avoid atomic defects during physical design seems promising at first glance, the resulting blacklist reduces the solution space substantially. Our evaluations (summarized later in Section V) show that this, in fact, makes it impossible to obtain circuits of relevant size. Thus, current design automation approaches are not applicable to realize practically-relevant SiDB circuits, and the development of a more sophisticated approach is imperative.

Motivated by that, this work proposes an alternative solution that does not aim at avoiding atomic defects but, instead, aims at designing specifically tailored gates on-the-fly which are capable of mitigating these defects—seamlessly integrating atomic defects as integral elements of their layout. The following outlines the general idea of the proposed method: 1) The SiDB surface is divided into hexagonal tiles, and a gate-level layout representation is determined based on the constraints imposed by the clocked tiles for a given netlist. 2) Each gate is matched to a dot-accurate SiDB implementation but calls the defect-aware method in an on-the-fly fashion whenever a defect prohibits the application of a gate from the standard library.

Example 4: Consider the scenario shown in Figure 3, where a tile contains neutral and negative atomic defects. Assuming that a previous placement and routing algorithm has designated this tile as an OR gate, it follows from Figure 3a that the *Bestagon* OR gate cannot be fabricated.



(a) The *Bestagon* OR [8] gate cannot be fabricated due to the neutral and the negative atomic defects.



(b) Tailored OR gate design for given atomic defects.

Fig. 3: Atomic defect conflict with the *Bestagon* OR gate and tailored SiDB OR gate design.

However, a solution is provided by the ability to design a tailored OR gate on-the-fly—seamlessly integrating atomic defects as integral elements of their layout as shown in Figure 3b.

IV. Defect-aware SiDB Circuit Design

To realize the general idea outlined above, we need: 1) an SiDB gate design method which realizes the desired functionality but, at the same time, is aware of all defects in the H-Si(100)-2×1 surface and 2) an overall physical design method which does the actual mapping but calls the defect-aware method in an on-the-fly whenever a defect prohibits the application of a gate from the standard library. Both contributions are described in the following.

A. On-the-fly Defect-Aware SiDB Gate Design

As mentioned in Section II-B, when designing an SiDB gate, the goal is to find the correct SiDB arrangement within the canvas so that the correct output is obtained for all input combinations.

Thus, the proposed algorithm receives as input an STM H-Si(100)-2×1 surface scan region S, with all atomic defects being stored, and a skeleton. Then, a given number of SiDBs are placed within the canvas while guaranteeing no collision with atomic defects. Afterward, the obtained SiDB layout is physically simulated for all 2^n input combinations in the presence of all atomic defects given by S, where n is the number of logical input pins. If a valid SiDB gate implementation is found for the given Boolean function, the design process is completed—otherwise, the whole process is repeated with a different arrangement of canvas SiDBs. Since in the worst case, many different layouts have to be simulated before a valid one is found, fast physical simulation is required to ensure fast gate design.

The details of the proposed algorithm are illustrated by the pseudocode Algorithm 1. It starts with a Boolean function f for which the SiDB gate implementation is to be created, an STM H-Si(100)-2×1 surface scan area S to place the gate, the gate skeleton K with canvas C, the number d of SiDBs which are placed in the canvas, and the physical simulation parameters P which are used for the simulation that determines if the gained SiDB layout (skeleton + canvas SiDB + atomic defects) fulfills the Boolean function. All possible arrangements of d SiDBs in the canvas C that do not collide

Algorithm 1: Defect-Aware SiDB Gate Design

Input: Boolean function $f : \mathbb{B}^n \to \mathbb{B}^m$ to implement
Input: STM H-Si(100)-2×1 surface scan area S to design the gate
Input: Gate skeleton K with canvas C
Input: Number of SiDBs d to place in the canvas
Input: Physical simulation parameters $P = \{\mu_-, \lambda_{tf}, \epsilon_r\}$
Output: SiDB gate that implements f with d canvas SiDBs at location S

```
1  D ← atomic defects in S
2  pos ← all arrangements of d SiDBs in C with pos ∩ D = ∅
3  foreach p ∈ pos do
4      G ← K ∪ D ∪ p
5      for i = 0 ... 2^n − 1 do
6          simresult ← simulate L with input pattern i given P
7          if L does not implement f(i) then
8              goto Line 4 and continue with next p
9          end if
10     end for
11     return G
12 end foreach
13 return ∅
```

with any defects are collected as pos in Line 2. Each of these arrangements is individually added to the given skeleton K together with the atomic defects D in Line 4. The resulting gate layout G is simulated in up to 2^n input configurations, where n is the number of inputs, to check if the correct output is obtained. If this is not the case for input configuration i, the whole process is repeated, starting with the next arrangement of SiDBs in the canvas in Line 3. Otherwise, i. e., if G implements f, the design process can be stopped and G is returned in Line 11.

B. Overall Physical Design Method

Having introduced and explained the on-the-fly and defect-aware SiDB gate design algorithm, the next step is to outline how it is combined with placement and routing to design SiDB circuits of significant size and complexity in the presence of atomic defects for the first time. While the general idea has already been explained in Section III-B, this section will focus on the implementation details. The whole idea is summarized by the pseudocode in Algorithm 2.

It starts with a circuit specification given as a gate-level netlist N, an STM H-Si(100)-2×1 surface scan area S to design the circuit, a gate skeleton library Lib_K, the number of SiDBs d to place in the canvas when designing SiDB gates, and the parameters P for the defect-aware SiDB gate design algorithm.

First, all defects are extracted from S and stored as D in Line 1. Subsequently, placement and routing is conducted in Line 2 which tries to place and route the gates in a way that the skeletons do not overlap or collide with neutral atomic defects. If no gate-level layout solution is found, then the given circuit cannot be designed under the given minimal number of constraints.

In the case where a gate-level layout is found, i. e., placement and routing were successful, the defect-aware SiDB gate design algorithm (Algorithm 1) is called. Line 7 enumerates all gates in the gate-level layout. For a given gate, first, the corresponding skeleton is chosen in Line 8 to define the pins, which is used as input for the defect-aware SiDB gate design algorithm together with the number of canvas SiDBs d, and the physical parameters P (Line 9). If a valid SiDB gate implementation can be designed, it is added to

Algorithm 2: Defect-Aware SiDB Circuit Design

Input: Circuit specification given as a gate-level netlist N
Input: STM H-Si(100)-2×1 surface scan area S to place the circuit
Input: Gate skeleton library Lib_K
Input: Number of SiDBs d to place in the canvas
Input: Physical simulation parameters P to pass to Algorithm 1
Output: Defect-aware dot-accurate SiDB layout that implements N

1 $D \leftarrow$ atomic defects in S
2 $L_G \leftarrow$ gate-level layout obtained by performing defect-aware placement and routing of N on S avoiding D using [26]
3 **if** *placement and routing was unsuccessful* **then**
4 | **return** $\emptyset$
5 **end if**
6 $L_{SiDB} \leftarrow \emptyset$
7 **foreach** *gate* $g \in L_G$ **do**
8 | $K_g \leftarrow$ appropriate skeleton for g from Lib_K
9 | $SiDB_g \leftarrow \text{GateDesign}(f(g), S, K_g, d, P)$ `// Algorithm 1`
10 | **if** *gate design was unsuccessful* **then**
11 | | blacklist gate g at this surface position
12 | | goto Line 2
13 | **end if**
14 | $L_{SiDB} \leftarrow L_{SiDB} \cup SiDB_g$
15 **end foreach**
16 **return** L_{SiDB}

the layout L_{SiDB} in Line 14. If not, it is blacklisted at this surface position in Line 11, and the placement and routing are repeated in Line 2 with this newly added and further constraint. After each gate was designed successfully as an SiDB structure, the dot-accurate layout is returned in Line 16.

However, to provide maximum speed of the design process, we implemented another feature which is not mentioned in Algorithm 2 for simplicity: since the gate design process consumes a considerable amount of runtime, it is desired to call the gate designer as few times as possible. Therefore, before an SiDB gate is designed in Line 9, it is checked if a corresponding implementation of the SiDB gate from a given pre-computed gate library can readily be used.

V. Experimental Evaluations

In an extensive experimental evaluation, we first demonstrate that simply avoiding all tiles where atomic defect conflicts occur—as suggested by the first proposed attempt in [26]—is an insufficient approach for designing SiDB circuits of relevant complexity and size. Second, however, it is shown that the proposed approach of designing tailored SiDB gates on-the-fly is promising. To this end, the experimental setup is described in Section V-A. Subsequently, the results obtained are presented in Section V-B. Finally, the results are discussed and analyzed in Section V-C.

A. Experimental Setup

For the experimental evaluation, a variety of different benchmark circuits [8] are designed on significantly defective surfaces. Walter *et al*. [26] conducted their evaluation using defective surfaces characterized by varying defect concentrations. These defective surfaces, which are publicly available, were utilized in our evaluation at defect concentrations of both $0.5\,\%$ and $1\,\%$. The proposed defect-aware SiDB circuit designer has been implemented in C++17 and was compiled with AppleClang 14.0.0. The design of all benchmark circuits was carried out on a macOS 13.0 machine with an Apple Silicon M1 Pro SoC with $32\,\text{GB}$ of integrated main memory. Because crossing wires are very sensitive to charged atomic defects [16], the placement and routing were conducted without crossing wires, enforcing planarity, wherever possible. *QuickExact* [14] is used as the physical simulator for the on-the-fly and defect-aware SiDB gate design algorithm due to its significantly higher precision and efficiency compared to other simulation engines. Support for atomic defect simulation has been added to *QuickExact* as proposed by Ng *et al*. [16].

To support open research, the implementation is available in the *fiction*[1] framework [29] which is part of the *Munich Nanotech Toolkit* (MNT, [30]).

B. Obtained Results

The results obtained from the design of a variety of benchmark circuits [8] are summarized in Table I. The first two columns contain the names of the benchmark circuits and the number of layout aspect ratios explored until a valid design was achieved using both the state-of-the-art approach (initial attempt [26]) and the proposed solution for realistic simulated surfaces with defect rates of $0.5\,\%$ and $1\,\%$ as presented by Walter *et al*. [26]. One additional columns per defective surface highlight the relative reduction in the number of enumerated aspect ratios of the state-of-the-art and the proposed approach.

C. Discussion

The simulation results show that it is impossible to design SiDB circuits of relevant complexity and size for significantly defective surfaces simply by avoiding all tiles where defect conflicts occur, as proposed by the first attempt [26]. This demonstrates first that atomic defects have a massive impact on physical design, and second that an alternative approach is imperative.

Designing the same circuits as before on a $1\,\%$ defective surface but using the proposed approach shows that the incorporated ideas have a significant positive impact on the circuit design. The proposed approach successfully designs almost all benchmark circuits except the last three (*majority*, *majority-5-r1*, *cm82a-5*). This trend is further confirmed by examining the simulation results for the $0.5\,\%$ defective surface, where the proposed approach provides solutions for all 12 circuits. Consequently, the proposed approach can design SiDB circuits of relevant size and complexity on significantly defective surfaces, marking a significant milestone in the advancement of SiDB circuits.

Moreover, the proposed approach achieves a remarkable reduction of up to $85.86\,\%$ in the number of aspect ratios that needed to be enumerated before a valid circuit design could be obtained. In other words, it significantly expands the scope for circuit design on defective SiDB surfaces compared to the state-of-the-art algorithm.

These observed qualities of the proposed approach underscore the benefits of the ideas introduced in this work. For the first time, SiDB circuits of relevant size and complexity can be designed on defective surfaces.

[1] https://github.com/cda-tum/fiction

TABLE I: Results for the state-of-the-art [26] (SOTA) and the proposed approach on simulated defective H-Si(100)-2×1 surfaces. The number of aspect ratios (#AR) describes how many different layout aspect ratios must be enumerated until an area is found where the circuit can be designed successfully in the presence of atomic defects.

Benchmark [31], [32]		1 % defective			0.5 % defective		
		SOTA [26]	Proposed	Δ #AR [%]	SOTA [26]	Proposed	Δ #AR [%]
Name	#AR	#AR	#AR		#AR	#AR	
xor2	8	588	89	−85.86	30	14	−53.33
xnor2	8	588	89	−85.86	30	14	−53.33
par_gen	9	—	96	—	56	22	−60.71
xor5_r1	17	—	140	—	95	32	−66.32
xor_majority	17	—	140	—	95	32	−66.32
mux21	20	—	92	—	—	39	—
par_check	46	—	195	—	—	79	—
t	63	—	341	—	—	79	—
t_5	63	—	341	—	—	150	—
c17	36	—	225	—	—	77	—
newtag	66	—	439	—	—	218	—
majority	83	—	426	—	—	127	—
majority_5_r1	76	—	—	—	—	220	—
cm82a_5	175	—	—	—	—	—	—

VI. Conclusion

In recent years, significant progress has been made in developing a complete physical design workflow for the *Silicon Dangling Bond* (SiDB) technology. However, H-Si(100)-2×1 surfaces, required for the fabrication of SiDB logic, are affected by a substantial amount of atomic defects [17], [18], [28]. Consequently, established design automation flows, which are defect-agnostic, are inadequate and have not kept pace with the latest experimental findings and advances in fabrication capabilities. A first attempt was recently proposed that extends established defect-agnostic physical design methods by rudimentary defect-aware capabilities. While promising at first glance, in this work, we show that this first attempt yields unsatisfactory results. Subsequently, we present a novel approach that automatically designs a tailored SiDB gate on-the-fly whenever an SiDB gate encounters atomic defects in its vicinity, thereby incorporating these atomic defects into its layout as an integral part.

Our experimental evaluations confirm that the proposed approach is capable of designing SiDB circuits of significant complexity and size, even in the presence of atomic defects for the first time.

To support open research and open data, the implementation and the simulation results are publicly available as part of the *Munich Nanotech Toolkit* (MNT, [30]).

References

[1] Nobel Prize. (2023) The nobel prize in chemistry 2023. Nobel Media AB. [Online]. Available: https://www.nobelprize.org/prizes/chemistry/2023/press-release/

[2] Cotta, M. A., "Quantum dots and their applications: What lies ahead?" *ACS Applied Nano Materials*, 2020.

[3] T. Huff *et al.*, "Binary Atomic Silicon Logic," *Nature Electronics*, 2018.

[4] T. R. Huff *et al.*, "Atomic White-Out: Enabling Atomic Circuitry through Mechanically Induced Bonding of Single Hydrogen Atoms to a Silicon Surface," *ACS Nano*, 2017.

[5] R. Achal *et al.*, "Lithography for Robust and Editable Atomic-scale Silicon Devices and Memories," *Nature Communications*, 2018.

[6] J. Wyrick *et al.*, "Atom-by-Atom Fabrication of Single and Few Dopant Quantum Devices," *Advanced Functional Materials*, 2019.

[7] J. Croshaw *et al.*, "Ionic Charge Distributions in Silicon Atomic Surface Wires," *Nanoscale*, 2021.

[8] M. Walter *et al.*, "Hexagons are the Bestagons: Design Automation for Silicon Dangling Bond Logic," in *DAC*, 2022.

[9] S. Hofmann *et al.*, "Scalable Physical Design for Silicon Dangling Bond Logic: How a 45° Turn Prevents the Reinvention of the Wheel," in *IEEE-NANO*, 2023.

[10] M. Walter *et al.*, "Design Automation for Field-coupled Nanotechnologies," in *ISVLSI*, 2020.

[11] S. Hofmann *et al.*, "Thinking Outside the Clock: Physical Design for Field-coupled Nanocomputing with Deep Reinforcement Learning," in *ISQED*, 2024.

[12] ——, "Late Breaking Results: Wiring Reduction for Field-coupled Nanotechnologies," in *DAC*, 2024.

[13] J. Drewniok *et al.*, "*QuickSim*: Efficient *and* Accurate Physical Simulation of Silicon Dangling Bond Logic," in *IEEE-NANO*, 2023.

[14] ——, "The Need for Speed: Efficient Exact Simulation of Silicon Dangling Bond Logic," in *ASP-DAC*, 2024.

[15] S. S. H. Ng *et al.*, "SiQAD: A Design and Simulation Tool for Atomic Silicon Quantum Dot Circuits," *TNANO*, 2020.

[16] ——, "Simulating Charged Defects in Silicon Dangling Bond Logic Systems to Evaluate Logic Robustness," *TNANO*, 2024.

[17] J. Croshaw *et al.*, "Atomic Defect Classification of the H-Si(100) Surface through Multi-mode Scanning Probe Microscopy," *Beilstein Journal of Nanotechnology*, 2020.

[18] T. R. Huff *et al.*, "Electrostatic Landscape of a Hydrogen-Terminated Silicon Surface Probed by a Moveable Quantum Dot," *ACS Nano*, 2019.

[19] N. Pavliček *et al.*, "Tip-Induced Passivation of Dangling Bonds on Hydrogenated Si(100)-2×1," *Applied Physics Letters*, 2017.

[20] M. Rashidi *et al.*, "Automated Atomic Scale Fabrication," US Patent, 2022, 17/429,443.

[21] R. Achal *et al.*, "Detecting and Directing Single Molecule Binding Events on H-Si (100) with Application to Ultradense Data Storage," *ACS Nano*, 2019.

[22] M. Taucer, "Silicon Dangling Bonds Non-equilibrium Dynamics and Applications," 2015.

[23] R. Wolkow *et al.*, "Initiating and Monitoring the Evolution of Single Electrons Within Atom-Defined Structures," 2023, uS Patent 11,635,450.

[24] R. A. Wolkow *et al.*, "Multiple Silicon Atom Quantum Dot and Devices inclusive thereof," US Patent 10 937 959, 2021.

[25] R. Wolkow *et al.*, "Initiating and Monitoring the Evolution of Single Electrons within Atom-defined Structures," US Patent 11 047 877, 2021.

[26] M. Walter *et al.*, "Atomic Defect-Aware Physical Design of Silicon Dangling Bond Logic on the H-Si(100)-2×1 Surface," in *DATE*, 2024.

[27] ——, "Placement & Routing for Tile-based Field-coupled Nanocomputing Circuits is $\mathcal{NP}$-complete," *JETC*, 2019.

[28] M. Rashidi *et al.*, "Deep learning-guided surface characterization for autonomous hydrogen lithography," *Machine Learning: Science and Technology*, 2020.

[29] M. Walter *et al.*, "*fiction*: An Open Source Framework for the Design of Field-coupled Nanocomputing Circuits," 2019.

[30] ——, "The Munich Nanotech Toolkit (MNT)," in *IEEE-NANO*, 2024.

[31] G. Fontes *et al.*, "Placement and Routing by Overlapping and Merging QCA Gates," in *ISCAS*, 2018.

[32] A. Trindade *et al.*, "A Placement and Routing Algorithm for Quantum-dot Cellular Automata," in *SBCCI*, 2016.