ParaMitE: Mitigating Parasitic CNFETs in the Presence of Unetched CNTs

Sanmitra Banerjee*, Arjun Chaudhuri*, Jinwoo Kim†, Gauthaman Murali†, Mark Nelson§, Sung Kyu Lim† and Krishnendu Chakrabarty*

*Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708, USA
†School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
§SkyWater Technology, Bloomington, MN 55425, USA

2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD) | 978-1-6654-4507-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICCAD51958.2021.9643513

***Abstract*—Carbon nanotube FETs (CNFETs) are emerging as an alternative to silicon devices for next-generation computing systems. However, imperfect carbon nanotube deposition during CNFET fabrication can lead to the formation of difficult-to-etch CNT aggregates in the active layer. These CNT aggregates can form parasitic CNFETs (para-FETs) that are modulated by adjoining gate contacts or back-end-of-line metal layers, thereby forming conditional shorts and stuck-at faults. We show that even weak (parametric) para-FETs can lead to a degraded static noise margin in CNFET-based design. We propose ParaMitE, a layout optimization method that horizontally flips selected standard cells *in situ* to minimize the number of para-FETs that can arise due to unetched CNTs. As we modify only the cell orientation (and not the cell placement), the impact on the power, timing, and wire length of the CNFET-based design is negligible. Simulation results for several benchmarks show that the proposed method can mitigate up to 60% of the possible para-FET locations (90% of the most critical locations) with only a 3% increase in the total wire length. ParaMitE can enable yield ramp-up at the foundry by providing guidance on which para-FETs can be avoided by design, and conversely, which CNT aggregates must be removed through processing steps.**

I. Introduction

Carbon nanotube field-effect transistors (CNFETs) are being explored as a promising successor to Si-MOSFETs in the post Moore's law era [1]. The near-ballistic carrier transport in carbon nanotubes (CNTs) enables high-drive current capability and low intrinsic delay in CNFETs [2]–[4]. Moreover, CNFETs exhibit excellent subthreshold swing and low leakage current, and are reconfigurable during runtime [5] [6]. However, several roadblocks need to be addressed before the high-volume manufacture of CNFETs becomes feasible. While test and mitigation techniques for mCNTs [7], misaligned CNTs [8], and process variations [9] show promising results, several other yield limiters are yet to be explored.

In a typical CNFET manufacturing flow, CNTs in the unwanted regions on the wafer are etched away using high-density oxygen plasma [10]. Although CNTs are deposited uniformly over the wafer, nanotubes can form aggregates due to the strong mutual Van der Waals force [11]. These CNT bundles are difficult to etch and remain as residual particles that limit die yield. These unetched CNTs can be turned on by an adjacent gate electrode or the back-end-of-line (BEOL) metal layer, thereby forming "parasitic" CNFETs or *para-FETs*.

In this paper, we show that even weak para-FETs can lead to stuck-at faults and significant degradation in the static noise margin. We propose a low-cost layout optimization method to mitigate this adverse impact. Our method horizontally flips selected cells *in situ* to ensure that unetched CNTs, even if present, will not lead to the formation of para-FETs. The main contributions of this paper are:

- Identification of the conditions under which unetched CNTs can lead to para-FETs;
- Evaluation of the performance impact of para-FETs;
- Para-FET mitigation in the presence of unetched CNTs (ParaMitE): An optimization technique to mitigate the impact of para-FETs with minimum impact on the power consumption, performance, and area (PPA) metrics.

As part of the pre-processing step for ParaMitE, we detect locations in the layout where unetched CNTs can lead to para-FETs (in linear/saturation regime); we call these locations *hotspots*. Simulation results over several CNFET-based benchmarks show that ParaMitE can reduce the number of hotspots by up to 60% (with up to 90% fewer catastrophic hotspots) with only 3% increase in wire length, 0.4% increase in the total power, and 2% increase in the critical path delay.

The remainder of the paper is organized as follows. Section II reviews CNFET fundamentals and the scenarios under which unetched CNTs can lead to para-FETs. Section III analyzes the cell-level impact of para-FETs. In Section IV, we present the ParaMitE optimization method and explore the different tunable parameters. The results obtained by applying ParaMitE to CNFET-based benchmarks and the impact on PPA are shown in Section V. We draw conclusions in Section VI.

II. Background and Motivation

A. Fundamentals of CNFETs

Single-walled semiconducting CNTs form the channels in CNFETs, and these channels are controlled by a gate contact. The p- and n-type doping of CNTs (for manufacturing p-type and n-type CNFETs) is typically carried out by metal work-function engineering of the source and drain contacts [10]. The source, drain, and gate contacts are patterned using low-

This research was supported in part by the DARPA ERI 3DSOC program under Award HR001118C0096.

978-1-6654-4507-8/21/$31.00 ©2021 IEEE
Authorized licensed use limited to: Georgia Institute of Technology. Downloaded on March 12,2022 at 13:12:22 UTC from IEEE Xplore. Restrictions apply.

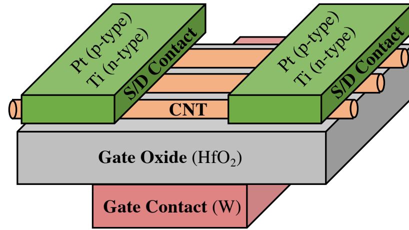


Fig. 1: 3D schematic of a bottom-gate CNFET: the S/D contacts use higher-work-function (lower-work-function) metal like platinum (titanium) for p-type (n-type) CNFET.

temperature lithographic steps that are compatible with the standard back-end-of-line (BEOL) fabrication flow.

A comparative analysis of the three CNFET geometries—gate-all-around (GAA), top-gate, and bottom-gate—has been carried out in [12]. The bottom-gate CNFET design involves a less complex fabrication process and achieves higher EDP benefits compared to the GAA and top-gate geometries. A 16-bit microprocessor with bottom-gate CNFETs has been demonstrated by a commercial semiconductor foundry [13]. A representative 3D schematic of a bottom-gate CNFET is shown in Fig. 1. Therefore, in this paper, we consider circuits made of bottom-gate CNFETs for the demonstration of ParaMitE.

The manufacturing process of bottom-gate CNFETs involves multiple stages of CNT deposition, etch, and contact patterning using multiple mask layers and silicon-compatible air-stable materials [13], [14]. After CNT deposition, unwanted CNTs outside CNFETs are etched away using an oxygen-plasma etch process [15]. A photoresist layer is used to protect the CNTs inside CNFETs from the etch process.

B. Parasitic CNFETs due to Unetched CNT Aggregates

Like any other manufacturing process, high-density oxygen-plasma etch is associated with inherent process variability [16]. CNTs may be left unetched due to a high density of CNTs (> 50 CNTs per μm [14]) and a non-ideal etch process [17].

The presence of CNT aggregate defects increases the chances of CNTs left unetched even by a mature etch process. Experimental evidence of CNTs getting bundled after deposition and forming aggregates have been reported in [10]. A technique called "RINSE" is used in [10] to reduce CNT aggregate density by adhesion coating of the deposited CNTs and subsequent sonification to agitate and remove the aggregates. However, RINSE may lead to a reduction in the CNT density through the inadvertent removal of non-aggregate CNTs. Moreover, RINSE requires a large power supply for providing megasonic power for the sonification of large commercial-scale wafers. High power consumption and a reduction in CNT density in the active layer are major drawbacks associated with RINSE. On the other hand, design or layout-level methods for minimizing the impact of potential CNT aggregates are more scalable, robust, and better suited for high-volume manufacturing, compared to fabrication-based approaches such as RINSE [18].

Fig. 2 shows several CNT aggregates in a die manufactured at a commercial semiconductor foundry[1]. Such aggregates are

[1]The foundry name is being withheld at this review stage due to ICCAD's double-blind submission requirement.

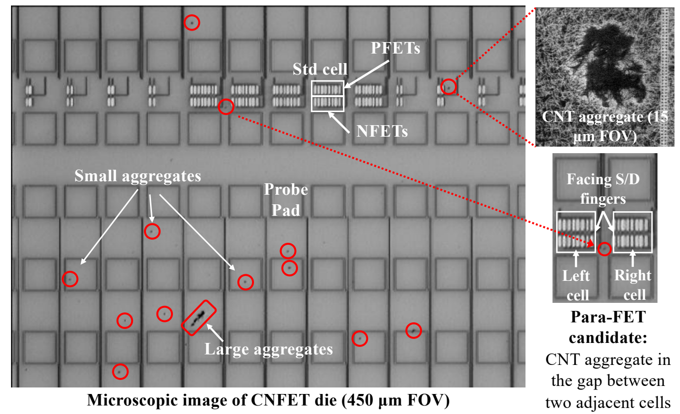


Fig. 2: SEM image of CNT aggregates in the gap between adjacent standard cells in a CNFET die manufactured at a commercial foundry ("FOV" stands for field-of-view).

difficult to etch away, often affecting subsequent steps in the fabrication flow and degrading the quality and yield of the product. The failure to etch aggregates results in the presence of unetched CNTs at undesired locations in the layout, i.e., outside CNFETs. The unetched CNTs may connect two nodes of the circuit leading to two possible consequences:

- If an unetched CNT between two nodes is metallic, the nodes get permanently shorted.
- If the unetched CNT is semiconducting, the nodes will be conditionally shorted, i.e., the unetched CNT will conduct current if there exists a signal-carrying interconnect in proximity that enables carrier transport in the CNT. We refer to such an unetched CNT driven by a proximal metal-routing layer as a para-FET.

The proposed ParaMitE framework achieves mitigation of para-FET impact by *in situ* horizontal flipping of cells with minimal power-performance-area (PPA) overhead. We prioritize mitigation of high-impact and high-likelihood para-FET hotspots in the layout. Some para-FET hotspots with weaker impact may be left unaddressed by ParaMitE. Even in such cases, the strength of the unaddressed hotspots (and hence, their impact) is minimized by ParaMitE. The locations of the unaddressed hotspots can also be provided to the foundry for targeted aggregate removal during fabrication.

C. Analysis of Parasitic CNFET Formation

The necessary conditions for the formation of para-FETs between adjacent standard cell instances in the layout are: (a) presence of unetched CNTs (or CNT aggregates) between the rightmost source/drain (S/D) contact (or finger) of left standard cell and leftmost S/D contact of the right standard cell; (b) some portion of at least one unetched CNT must be covered by a metal layer used for routing. The metal layer acts as the controlling gate for the para-FET. Note that, larger the gap between adjacent cells and smaller the height of individual cells, lower is the likelihood of occurrence of unetched CNTs that extend from the left cell to the right.

The para-FET strength depends on: (a) gap between the adjacent cells, which determines the length of the para-FET; (b) portion of unetched CNTs covered by the metal layer, which

Authorized licensed use limited to: Georgia Institute of Technology. Downloaded on March 12,2022 at 13:12:22 UTC from IEEE Xplore. Restrictions apply.

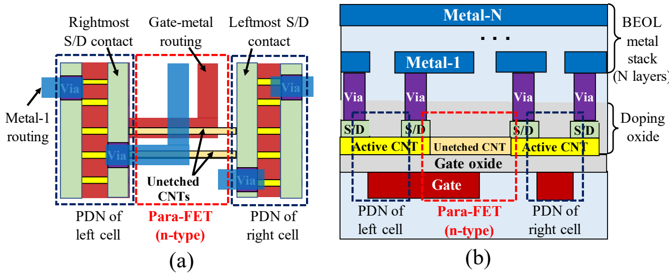


Fig. 3: Para-FET formation due to unetched CNTs between PDNs of adjacent cells: (a) top view and (b) side view. The gate oxide and doping oxide are not shown in (a) for visual clarity.

determines the effective gate area of the para-FET; (c) logical connections of the facing S/D contacts of the adjacent cells. If both facing contacts are tied to the power supply (in pull-up network or PUN) or ground (in pull-down network or PDN), the formed para-FET will be in cut-off, i.e., inactive, and not affect performance, as its two ends are shorted. Therefore, smaller gaps between cells, high-density metal routing, and cell pairs with at least one of the facing S/D contacts in PUN (PDN) not connected to the power supply (ground) aggravate the potential impact of para-FETs on circuit performance.

In summary, the factors determining the formation and impact of active para-FETs are: (a) standard cell orientations in the layout; (b) density of CNT aggregates after CNT deposition; (c) efficiency of the CNT etching mechanism and RINSE; (d) density of metal routing closest to the active layer (i.e., the CNT-deposited plane); (e) standard cell height; and (f) the gap between adjacent standard cells. The nature of para-FETs formed between PUNs (PDNs) of adjacent cells is p-type (n-type). Fig. 3 illustrates the side and top views of an n-type para-FET between the PDNs of adjacent standard cells. The probability of para-FET occurrence between PUNs and PDNs of adjacent standard cells in a layout is analyzed in [17].

III. Cell-Level Impact of Para-FETs

Para-FETs can form conditional shorts between the terminals of two horizontally adjacent cells in a CNFET-based design layout. Consider two unit-sized inverters with unetched CNTs aggregated in the active layer between their PDNs (inset in Fig. 4). In the presence of an overlying Metal-1 (or underlying gate metal), the unetched CNT can form an n-type para-FET that shorts the output of the left inverter, $Y1$, to VSS. The strength of this para-FET depends on the gap between the neighboring cells, the metal layer driving the para-FET, and the size of the CNT aggregate. Similarly, p-type para-FETs are formed due to unetched CNT aggregates in the PUN.

The width of the unetched CNT aggregate covered by the metal layer determines the gate width of the para-FET. The channel length (L_{para}) is given by the gap between the adjacent contacts of the two cells. Para-FETs driven by the gate layer will be significantly stronger compared to a similarly sized para-FET driven by Metal-1. This is due to the higher dielectric constant of the gate dielectric HfO_2 (compared to the inter-layer dielectric SiO_2) and a lower oxide thickness of 15 nm (compared to 360 nm for the Metal-1 layer).

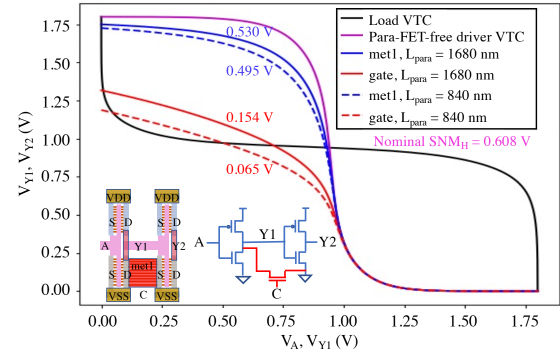


Fig. 4: Voltage transfer curves (VTCs) of a cascaded inverter pair in the presence of para-FETs shorting the PDN. The respective high static noise margin (SNM_H) value is listed alongside each VTC.

TABLE I: SNM_H (V) of a cascaded inverter pair in the presence of para-FETs shorting the PDN. The nominal SNM_H in the absence of para-FETs is 0.608 V.

Metal Layer Type	L_{para} (nm)				
	840	1680	2520	3360	4200
Metal-1	0.495	0.530	0.551	0.563	0.571
Gate	0.065	0.154	0.212	0.253	0.285
Metal-1, Gate	0.064	0.153	0.211	0.252	0.284

In Fig. 4, we show the voltage-transfer curves (VTCs) of an inverter in the presence of para-FETs with L_{para} equal to 840 nm and 1680 nm in the PDN. To present the worst-case impact, we assume that the width of the CNT aggregate is equal to the width of a nominal CNFET in the PDK, and the entire aggregate is covered by the Metal-1 (or gate) layer. As the para-FET shorts the output to VSS, the pull-up performance of the inverter is affected, and the VTC deviates from the para-FET-free case when the input V_A is low. In the deviated VTC, the output-high voltage and the high static noise margin (SNM_H) are degraded. Observe that for the same L_{para}, the impact on the inverter pull-up operation is considerably higher when the para-FET is driven by the gate compared to when it is driven by Metal-1. Also, the strength of the para-FET (and its impact) decreases as its L_{para} increases.

Table I shows the SNM_H of the cascaded inverter-inverter pair under para-FETs driven by either Metal-1, or the gate layer, or both for different values of L_{para}. The impact of para-FETs driven by the gate or both Metal-1 and gate is catastrophic compared to those driven solely by Metal-1. Especially for strong gate-driven para-FETs ($L_{para} = 840$ nm), the negligible SNM_H denotes that the output is almost stuck at low. These simulation results demonstrate that strong para-FETs can lead to stuck-at faults while weak para-FETs can degrade the noise immunity of affected cells. In the next section, we propose an optimization technique that mitigates the majority of potential para-FET hotspots at negligible cost.

IV. Para-FET Mitigation in the Presence of Unetched CNTs: ParaMitE

Unetched CNTs between two adjacent cells in a row can form active para-FETs only if the rightmost S/D of the left cell and the leftmost S/D contact of the right cell are at different potentials. To mitigate the impact of para-FET hotspots, we horizontally flip selected cells in each row to ensure that this necessary condition for para-FET formation is not satisfied.

Authorized licensed use limited to: Georgia Institute of Technology. Downloaded on March 12,2022 at 13:12:22 UTC from IEEE Xplore. Restrictions apply.

A. Problem Formulation

As the CNTs are deposited horizontally in the active layer, para-FETs can only form between two horizontally adjacent cells in a row. Let C_i and C_{i+1} constitute a horizontally adjacent cell pair with a gap of $d_{i,i+1}$ between them and with C_i on the left. Under horizontal flips, the set of possible orientations of the cell pair is given by the ordered quadruple $O_{i,i+1} = ((C_i,\, C_{i+1}),\, (C_i^{'},\, C_{i+1}),\, (C_i,\, C_{i+1}^{'}),\, (C_i^{'},\, C_{i+1}^{'}))$. In this case, $C_i^{'}$ is the horizontally flipped form of C_i. Let $F_{i,i+1}^{k_i,PUN}$ ($F_{i,i+1}^{k_i,PDN}$) be a binary variable that is 1 if the rightmost S/D contact in the PUN (PDN) of the left cell and the leftmost S/D contact in the PUN (PDN) of the right cell are at different potentials, and is 0 otherwise. Here, $k_i \in \{0, 1, 2, 3\}$ and it denotes that the cells are in the k_i^{th} orientation in $O_{i,i+1}$. We then define the para-FET susceptibility of this cell pair, $p_{i,i+1}^{k_i}$, in the k_i^{th} orientation to be: $p_{i,i+1}^{k_i} = (1/d_{i,i+1}) \cdot \left(F_{i,i+1}^{k_i,PUN} + F_{i,i+1}^{k_i,PDN}\right)$. Both the probability of CNTs remaining unetched and the strength of the resultant para-FET increases with decreasing $d_{i,i+1}$; therefore, $d_{i,i+1}$ is kept in the denominator in $p_{i,i+1}^{k_i}$.

For a row R with N cells, the row para-FET susceptibility is defined as $P_R = \sum_{i=1}^{N-1} p_{i,i+1}^{k_i}$. The global para-FET susceptibility of a CNFET design is then given by $G = \sum_R P_R$. Let ΔW, W^*, and τ denote the change in wire length, the maximum allowed total wire length, and the maximum allowed cell-pair para-FET susceptibility, respectively. The ParaMitE optimal cell flipping (ParaMitE-OCF) problem is given by:

PARAMITE-OCF

Instance: A layout with initial wire length W, and the para-FET susceptibility of each cell pair $p_{i,i+1}^{k_i} \forall i : i \in \mathbb{N} \wedge i \le N-1$ and $\forall k_i : k_i \in \{0, 1, 2, 3\}$.

Objective: Find the optimal value of k_i (given by k_i^*) to minimize P_R such that: (1) the final wire length, $W + \Delta W \le W^*$; (2) $\forall i : i \in \mathbb{N} \wedge i \le N - 1, p_{i,i+1}^{k_i^*} \le \tau$.

Consider the decision version of a restricted form of the ParaMitE-OCF problem where $\tau = \infty$ and $p_{i,i+1}^{k_i} = 0\ \forall i : i \in \mathbb{N} \wedge i \le N-1, \forall k_i : k_i \in \{0, 1, 2, 3\}$.

R-PARAMITE-OCF

Instance: A layout with initial wire length W.

Question: Is there an orientation k_i such that the final wire length $W + \Delta W \le W^*$?

The problem of finding an optimal set of flips to minimize the total wire length in the layout (WL-OCF) is equivalent to R-ParaMitE-OCF and is known to be NP-complete [19]. In the following, we prove using the method of restriction [20] that because WL-OCF is NP-complete, ParaMitE-OCF is NP-hard. We also show that the decision-problem version of ParaMitE-OCF is in NP, therefore, it is NP-complete.

Theorem 1. *The decision-problem version of ParaMitE-OCF is NP-complete.*

Proof. Given a certificate k_i^*, the verification involves polynomial-time computation using G, $W + \Delta W$, and $p_{i,i+1}^{k_i^*}\ \forall i : i \in \mathbb{N} \wedge i \le N-1$. As the size of the certificate k_i^* and the verification are both polynomial in N, ParaMitE-OCF is in NP.

R-ParaMitE-OCF (which is a restricted form of ParaMitE-OCF) is equivalent to WL-OCF. Therefore, to solve an arbitrary instance of WL-OCF, we need to apply the following restrictions to any arbitrary instance of ParaMitE-OCF: 1) $\tau = \infty$, and 2) $p_{i,i+1}^{k_i} = 0\ \forall i : i \in \mathbb{N} \wedge i \le N-1, \forall k_i : k_i \in \{0, 1, 2, 3\}$. Therefore, solving ParaMitE-OCF must be at least as hard as solving WL-OCF. As WL-OCF is NP-complete, ParaMitE-OCF is NP-hard. As ParaMitE-OCF is in NP and is NP-hard, it is NP-complete. □

B. Integer Linear Programming Model

To solve ParaMitE-OCF, we harness the implicit enumeration capabilities of an integer-linear programming (ILP) model. For a row R_m, the row para-FET susceptibility P_{R_m} is independent of $P_{R_n}\ \forall m \neq n$. Thus to minimize the global susceptibility $G = \sum_R P_R$, we separately minimize each P_R. For a row R with N cells, we define binary decision variables $X_R = \{x_1, x_2, \ldots, x_N\}$. For a cell C_i, $x_i = 1$ if it is flipped and is 0 otherwise. For the ordered quadruple of possible orientations of adjacent cells, given by $O_{i,i+1}$, the values of $(x_i,\, x_{i+1})$ are (0, 0), (1, 0), (0, 1), and (1, 1), respectively. Recall that $p_{i,i+1}^{k_i}$ denotes the cell-pair para-FET susceptibility of the adjacent cells when they are in the k_i^{th} orientation in $O_{i,i+1}$ ($k_i \in \{0, 1, 2, 3\}$). The optimization problem for minimizing P_R is given by:

$$\underset{X_R}{\text{minimize}} \sum_{i=1}^{N-1} (1-x_i)(1-x_{i+1})p_{i,i+1}^{k_i=0} + x_i(1-x_{i+1})p_{i,i+1}^{k_i=1} + (1-x_i)x_{i+1}p_{i,i+1}^{k_i=2} + x_i x_{i+1} p_{i,i+1}^{k_i=3} \quad (1a)$$

subject to

$$W + \Delta W \le W^*, \quad (1b)$$

$$(1-x_i)(1-x_{i+1})p_{i,i+1}^{k_i=0} \le \tau \qquad i = 1, \ldots, N-1, \quad (1c)$$

$$x_i(1-x_{i+1})p_{i,i+1}^{k_i=1} \le \tau \qquad i = 1, \ldots, N-1, \quad (1d)$$

$$(1-x_i)x_{i+1}p_{i,i+1}^{k_i=2} \le \tau \qquad i = 1, \ldots, N-1, \quad (1e)$$

$$x_i x_{i+1} p_{i,i+1}^{k_i=3} \le \tau \qquad i = 1, \ldots, N-1 \quad (1f)$$

Note that $\forall i$, constraints 1c-1f are non-linear due to the $x_i x_{i+1}$ product term. Therefore, to linearize our model, we replace $x_i x_{i+1}$ with another binary variable y_i. This substitution necessitates two additional constraints for each i: (1g) $x_i + x_{i+1} - y_i \le 1$ and (1h) $x_i + x_{i+1} - 2y_i \ge 0$. Thus, for a row with N cells, the total number of decision variables is $2N-1$ (Nx_i's and $(N-1)y_i$'s) and the total number of constraints is $6N-5$ (1b and 1c-1h for $i = 1, \ldots, N-1$). Our model is computationally efficient because the number of decision variables and constraints are polynomial in N.

C. Considerations in ILP Modeling

1) Minimum Admissible τ for Solution to ILP Problem

During the ILP-based optimization of a row of cells, when no constraint is placed on the wire length, i.e., $W^* = \infty$ in

Authorized licensed use limited to: Georgia Institute of Technology. Downloaded on March 12,2022 at 13:12:22 UTC from IEEE Xplore. Restrictions apply.

constraint (1b), it is possible to compute, in polynomial time, the theoretical lower bound of τ, τ_{min}, for which at least one solution to the ILP problem exists. Note that an upper bound on τ, τ_{max}, is the maximum para-FET susceptibility across all possible cell-pair orientations in a row, for any arbitrary value of W^*. Such estimates of the lower and upper bounds provide a well-defined range in which τ can be tuned.

To compute τ_{min} for a row, we build a "consistency graph" CG to represent possible combinations of legal cell placement in the row. A vertex v in CG represents one of the four cell-pair orientations: $\{(C_i, C_{i+1}), (C_i^{'}, C_{i+1}), (C_i, C_{i+1}^{'}), (C_i^{'}, C_{i+1}^{'})\}$. Let v be represented by an ordered pair (v_l, v_r). Here, v_l and v_r denote the left and right cells (in given orientations) in an adjacent cell pair, respectively. For example, v_l (v_r) can represent C_i (C_{i+1}). A directed edge is present from vertex u to vertex v if $u_r = v_l$. Therefore, CG is a directed acyclic graph which, upon levelization, is seen to contain four vertices in a level l; $1 \leq l \leq N-1$ where N is the number of cells in a row. The four vertices in l correspond to the four possible cell-pair orientations: $\{(C_l, C_{l+1}), (C_l^{'}, C_{l+1}), (C_l, C_{l+1}^{'}), (C_l^{'}, C_{l+1}^{'})\}$. There are no edges between vertices in the same level in CG. Every vertex has exactly two fan-in edges and two fan-out edges, except the ones in level $l = 1$ ($l = N-1$) that have no incoming (outgoing) edges. Each vertex in CG has a weight that equals the corresponding para-FET susceptibility $p_{l,l+1}^{k_l}$. Fig. 5 illustrates CG for a row with four cells, with the leftmost (rightmost) cell denoted by C_1 (C_4).

Every path from $l = 1$ to $l = N-1$ represents a legal cell-row placement (with or without cell flips). Having the τ-constraint in the ILP model is equivalent to pruning CG by removing vertices with weights exceeding τ. Therefore, during ILP-based optimization under τ-constraint with $W^* = \infty$, the feasible solution space, i.e., possible cell-flip solutions, comprises the remaining paths from $l = 1$ to $l = N-1$ after pruning CG. Our objective is to find τ_{min} for which at least one path exists from $l = 1$ to $l = N-1$ in CG.

Fig. 6 presents the algorithm for computing τ_{min}. The inputs to the algorithm are CG and an array PC consisting of the susceptibility based weights of the vertices in CG. The `Levelize()` function levelizes CG to return its levels. Every vertex v in a level l ($2 \leq l \leq N-1$) has two incoming edges from vertices in $l-1$. As we traverse CG from $l = 1$ to $l = N-1$, the susceptibility weight $PC[v]$ is updated to $PC^*[v]$ for every vertex v such that $PC^*[v]$ is the minimum τ for which at least one path exists from $l = 1$ to v. The $PC[v]$ for all v in level l is updated

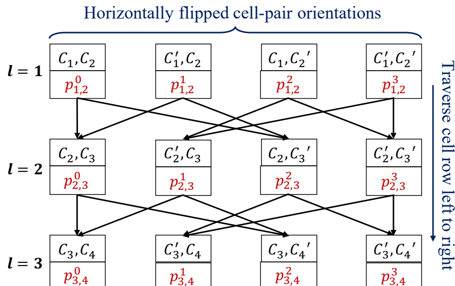


Fig. 5: Consistency graph for horizontal flipping of cells in a row.

Input: Consistency graph CG, para-FET susceptibility array PC
Output: Minimum feasible τ (τ_{min}) for $W^* = \infty$
$L \leftarrow$ `Levelize`(CG); //get levels (with respective vertices) in CG
$l_{max} \leftarrow$ final (or deepest) level in CG;
for $l \in L$ **do**
  for *vertex* $v \in l$ **do**
    $(f_1, f_2) \leftarrow$ fan-in vertices of v;
    $\tau_{path} \leftarrow \min(PC[f_1], PC[f_2])$;
    $PC[v] \leftarrow \max(PC[v], \tau_{path})$;
  end
end
$\tau_{min} \leftarrow \min_{\forall v \in l_{max}} PC[v]$;
return τ_{min}

Fig. 6: Pseudo-code to find minimum feasible τ for the ILP model when $W^* = \infty$, i.e., no constraint is placed on the wire length.

as: $PC^*[v] = \max(PC[v], \min(PC^*[f_1], PC^*[f_2]))$, where f_1 and f_2 are the fan-in vertices of v. This update policy guarantees that $PC^*[v]$ stores the minimum τ for which at least one path exists from level 1 to vertex v in level l and v is a part of that path. If we set $\tau < \min_{\forall v \in l} PC^*[v]$, all vertices in l are removed during pruning of CG resulting in no path from level 1 to $N-1$ and, consequently, no feasible cell-flip solution to the ILP problem. Therefore, for a solution to exist to the row-optimization problem under $W^* = \infty$, $\tau \geq \tau_{min} = \min_{\forall v \in l = N-1} PC^*[v]$. The time complexity of the proposed algorithm is $\mathcal{O}(N)$.

2) Relaxed Version of Proposed ILP Model

To precisely calculate the change in wire length for an X_R (1b), we need to reroute the interconnects in the entire layout. This can lead to a significant increase in the overall computation time because routing is computationally expensive [21] and needs to be performed several times to minimize P_R. Therefore, instead of limiting the wire length, we consider a relaxed form of constraint (1b) where we limit the number of cells flipped in a row. In a row R with N cells, the new constraint (1b) (in the relaxed model) is therefore given by $\sum_{i=1}^{N} x_i \leq F_R$; here F_R denotes the maximum number of cells in row R that can be flipped. While the wire length is not explicitly constrained in the relaxed model, simulation results on multiple benchmarks (see Section V.C) show negligible increase in wire length after applying ParaMitE. An alternative way to limit the number of cells flipped can be to add a Lagrangian regularization term (e.g., $\lambda(\sum_{i=1}^{N} x_i - F_R)$) to the objective function [22]. However, the Lagrangian-regularized model cannot always guarantee that the condition $\sum_{i=1}^{N} x_i \leq F_R$ will be satisfied. Therefore, we keep the flip count as a constraint (and not a regularization term). Note that we provide the total number of allowed global flip counts in the entire layout, F_G, as the input to ParaMitE. This value (F_G) is then distributed among the rows and the row budget, F_R, is determined using one of three budgeting policies as described next.

3) Assignment of Flip-Count Budgets for Row Optimization

We present three budgeting schemes for allocating the flip-count threshold F_R to a cell row in ParaMitE:

1) **Row-density (RD)**: The flip-count budget F_R assigned to a cell row with N cells is: $F_R = \frac{N}{N_G} \cdot F_G$, where N_G is the

Authorized licensed use limited to: Georgia Institute of Technology. Downloaded on March 12,2022 at 13:12:22 UTC from IEEE Xplore. Restrictions apply.

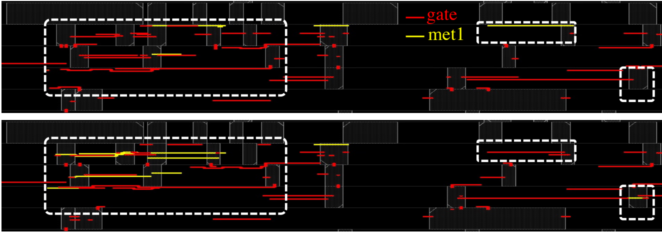


Fig. 7: Gate (red) and Metal-1 (yellow) layer on a region in the LDPC layout before (top) and after (bottom) ParaMitE. White dashed rectangles highlight the changes in the metal cover. The length of the gate (Metal-1) layer decreases (increases) post ParaMitE.

total number of cells in the design. The rows are traversed starting from the top of the layout during ParaMitE.

2) **Susceptibility-based (SB)**: The flip-count budget F_R assigned to a cell row is: $F_R = \frac{P_R}{P_G} \cdot F_G$, where P_R (P_G) is the sum of para-FET susceptibilities of cell pairs in the row (entire layout) before initiating ParaMitE. The rows are traversed starting from the top of the layout.
3) **Greedy row traversal with carryover (GR)**: This scheme uses the susceptibility-based scheme for allocating flip-count budgets to the rows before initiating ParaMitE. During ParaMitE execution, the row-wise cell-flipping is carried out in a greedy manner by traversing the rows in descending order of their para-FET susceptibilities, i.e, the row with the highest total susceptibility is optimized first. If a row ends up using fewer cell flips than the allocated budget, the remainder is carried over to the budget allocated to the next row in the order of traversal.

The RD budgeting scheme assigns F_R based on the cell count for each row; therefore, it is suitable for layouts where all rows have similar row susceptibility and strong para-FETs are not clustered. However, in this scheme, two rows with the same cell count but with significantly different P_R are assigned the same flip budget. Thus, when stronger para-FETs are clustered in rows with fewer cells, the RD scheme can lead to a majority of strong para-FETs remaining unmitigated. The SB budgeting addresses this issue by assigning higher flip budgets to rows with higher P_R. When the likelihood of unetched-CNT occurrence is high due to process immaturity, the GR scheme is recommended to reduce para-FET susceptibility further (beyond what the SB scheme achieves) by ensuring full utilization of the flip-count budget, F_G.

4) Presence of Metal Layers Driving Para-FETs

The cell-level impact of para-FETs depends on the metal layer driving the unetched CNTs (Sec. III). However, the metal cover on a row can change when cells in a different row are flipped and the interconnects are rerouted. Fig. 7 illustrates this change for a row in the LDPC benchmark before and after ParaMitE. Accurate consideration of the metal cover on a row is infeasible as the interconnect routing can change after the row has been optimized using ParaMitE. At the minimum, determining P_R will necessitate rerouting after each row (ideally, after each para-FET hotspot in a row) is handled; this is computationally expensive [23]. Therefore, we consider the worst-case scenario where all unetched CNTs are covered by a proximal metal layer and form para-FETs. Our pessimistic approach ensures that we are able to include all the para-FET candidates in the optimization framework.

V. Experimental Results

We evaluate the performance of ParaMitE on four open-source benchmarks — AES (50 MHz, 150K cells), OpenPiton (12.5 MHz, 274K cells), LDPC (33.33 MHz, 80K cells), and Nova (16.67 MHz, 184K cells), each synthesized using a 90 nm CNFET PDK from a commercial foundry.

A. Experimental Setup

In Section IV.C, we showed that the parameters in our ILP model can be selected in several ways— F_G can be assigned based on one of the three budgeting policies (row-density, susceptibility-based, or greedy+carryover) while τ can either be assigned globally to all rows (global mode) or its minimum value for each row can be considered (min mode). We consider the performance of ParaMitE under all these different settings. Fig. 8 shows our simulation framework. Using the layout obtained from the place-and-route (PnR) step in the RTL-to-GDS flow, we calculate the initial global para-FET susceptibility (G) and identify possible para-FET hotspot locations. We then apply ParaMitE (with appropriate F_G and τ) such that G and the overhead (increase in total power and wire length) are within acceptable limits. We implement the ParaMitE flow using the Python API of the Gurobi ILP solver and use Cadence Tempus 18.1 and Innovus 17.1 for PPA computation. Simulations were performed on two 2.53 GHz Intel E5630 Xeon CPUs with 64 GB RAM. The maximum ParaMitE run time (across all four benchmarks) is ~5 minutes, thereby the proposed solution is computationally efficient.

B. ParaMitE Evaluation

Fig. 9 shows the performance of ParaMitE under different F_G and budgeting policies. For all benchmarks, we observe that for some $F_G = F_G^*$, the reduction in hotspot count (RiH) and the reduction in global susceptibility (RiS) all saturate at some value of F_G. This saturation point signifies that all para-FET hotspots, which can be mitigated using *in-situ* horizontal flipping, have been mitigated. The percentage of cells actually flipped (CAF) also saturates at some F_G, but this saturation point can differ from F_G^*. Note also that for all F_G, RiS $\geq$ RiH; this is because we preferentially mitigate para-FET hotspots

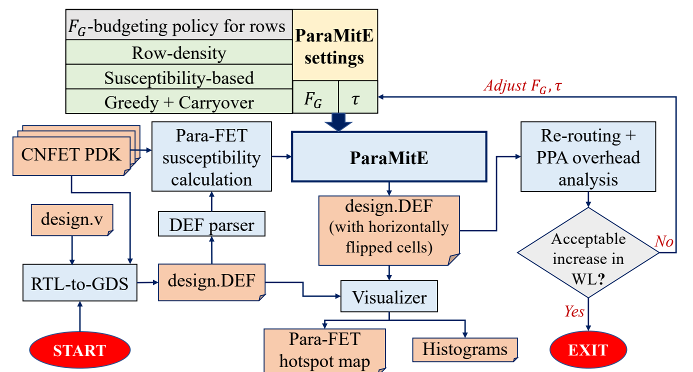
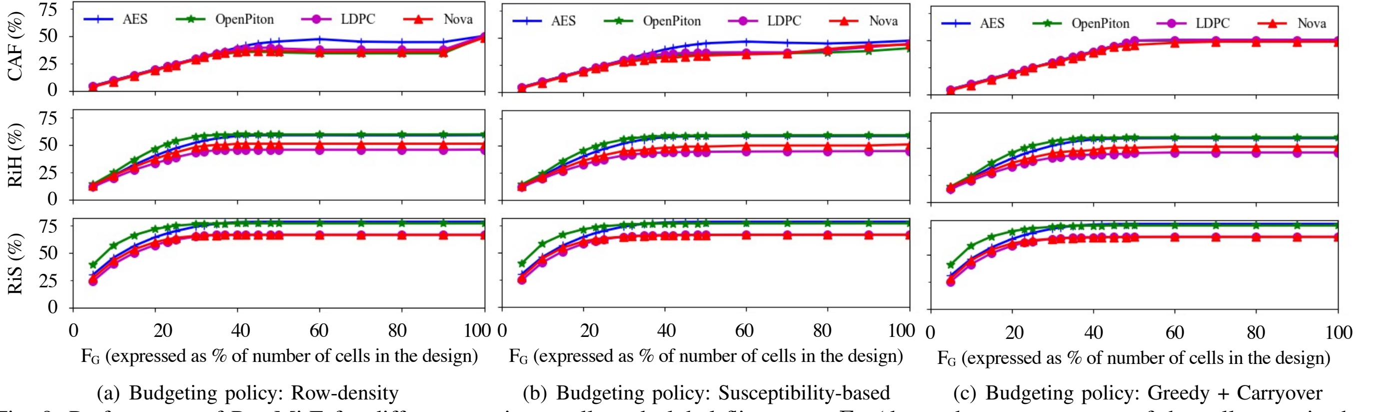


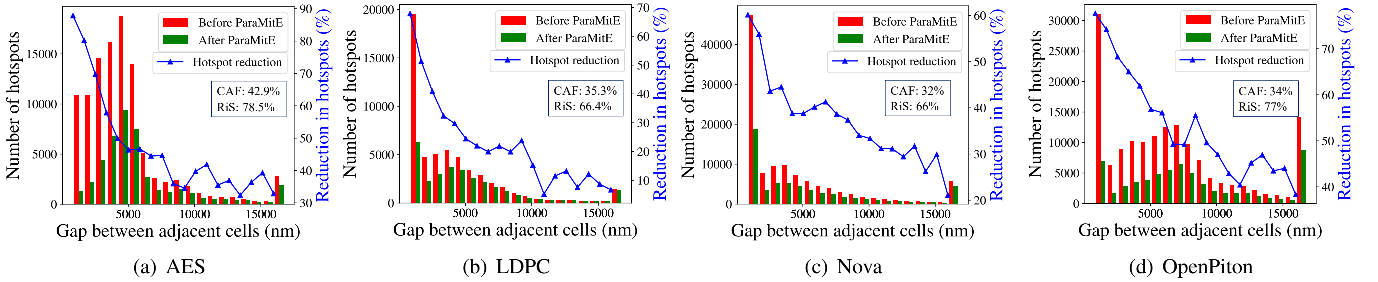
Fig. 8: ParaMitE framework for para-FET mitigation. The Design Exchange Format (DEF) file specifies the physical layout of an IC.

Authorized licensed use limited to: Georgia Institute of Technology. Downloaded on March 12,2022 at 13:12:22 UTC from IEEE Xplore. Restrictions apply.



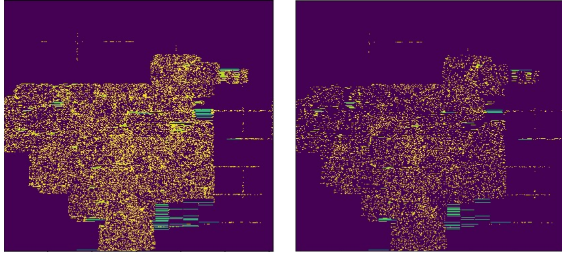
(a) Budgeting policy: Row-density (b) Budgeting policy: Susceptibility-based (c) Budgeting policy: Greedy + Carryover

Fig. 9: Performance of ParaMitE for different maximum allowed global flip count, F_G (denoted as a percentage of the cell count in the design, N_G) for different flip budget settings. CAF (Cells actually flipped) denotes the percentage of cells in the design that are flipped after applying ParaMitE. RiH (RiS) denotes the percentage reduction in hotspot count (percentage reduction in susceptibility) post-ParaMitE.



(a) AES (b) LDPC (c) Nova (d) OpenPiton

Fig. 10: Histograms of para-FET hotspots in the four benchmark circuits and the reduction in hotspot count after applying ParaMitE.



(a) Before ParaMitE (b) After ParaMitE

Fig. 11: Heat map showing para-FET hotspots in OpenPiton before and after applying ParaMitE (with susceptibility-based budgeting). Warmer (cooler) colors represent strong (weak) para-FET hotspots.

where the adjacent cells are closely placed (these hotspots have a higher contribution to the global susceptibility). Across the budgeting policies, for $F_G = F_G^*$, ParaMitE mitigates between 45%-60% of para-FET hotspots for different benchmarks (RiH plot in Fig. 9). Similarly, the para-FET susceptibility is reduced by 66%-77% from its initial value. We also observe that the susceptibility-based budgeting policy performs marginally better compared to the other policies.

Fig. 10 compares the frequency distribution of para-FET hotspots before and after ParaMitE using the susceptibility-based budgeting policy at $F_G = F_G^*$. We preferentially mitigate strong (small-gap) para-FETs; due to this, the reduction in count is maximum (up to 90%) for such para-FETs. Fig. 11 shows that there are considerably fewer para-FET hotspots in OpenPiton after ParaMitE (susceptibility-based budgeting) is applied. Similar results are obtained for other benchmarks.

Tuning τ can be used to limit the maximum cell-pair susceptibility in a row, for layouts where strong para-FET hotspots are clustered. A strict constraint on τ (e.g., τ_{min}) can also be used when the likelihood of CNTs remaining unetched is high. In other cases (uniformly distributed strong hotspots and an efficient etch process), τ_{min} and τ_{max} will be proximal and any feasible $\tau_{min} \leq \tau \leq \tau_{max}$ can be used for a row. Under a relaxed flip count budget (high F_G), we observe negligible change in the performance of ParaMitE for different values of τ (see Table II). For all other simulation results (in Fig. 9 and Fig. 10), we set τ to τ_{max} for all rows.

With ParaMitE, the number of para-FET hotspots in a layout is minimized; this leads to lower potential stuck-at faults and leakage power in the presence of unetched CNTs. However, horizontal flipping of cells can lead to an increase in the wire length (and, in turn, the total power consumption, and critical path delay). To analyze this trade-off, we perform eight simulation experiments for each benchmark:

1) F_G = 100% of N_G, RD budgeting, $\tau = \tau_{max}$ for each row;
2) F_G = 100% of N_G, RD budgeting, $\tau = \tau_{min}$ for each row;
3) $F_G = F_G^*$, RD budgeting, $\tau = \tau_{max}$ for each row;
4) $F_G = F_G^*$, SB budgeting, $\tau = \tau_{max}$ for each row;
5) $F_G = F_G^*$, GR budgeting, $\tau = \tau_{max}$ for each row;
6) F_G set to minimum value for which model is feasible, RD budgeting, $\tau = \tau_{min}$ for each row;
7) F_G set to minimum value for which model is feasible, SB budgeting, $\tau = \tau_{min}$ for each row;
8) F_G set to minimum value for which model is feasible, GR budgeting, $\tau = \tau_{min}$ for each row;

Table II shows the simulation results for these experiments.

Authorized licensed use limited to: Georgia Institute of Technology. Downloaded on March 12,2022 at 13:12:22 UTC from IEEE Xplore. Restrictions apply.

TABLE II: Simulation results showing the performance of ParaMitE for different F_G, τ, and budgeting policies. The first letter in Exp denotes the benchmark: A-AES, O-OpenPiton, L-LDPC, N-Nova. The second letter denotes the experiment number (defined in Section V.B). RD (row-density), SB (susceptibility-based), and GR (greedy+carryover) denote the three budgeting policies.

Exp	ParaMitE Settings		Before ParaMitE			After ParaMitE			ParaMitE Overhead (%)			
	F_G (% of N_G)	τ	#hotspots	Global suscep.	#Conditional stuck-at faults	#hotspots	Global suscep.	#Conditional stuck-at faults	Cells flipped	Total power	Critical path delay	Wire length
A1	100, RD	τ_{max}	106531	**60.80**	100	43832	**12.97**	15	50.56	+0.15	+1.81	+3.76
A2	100, RD	τ_{min}	106531	**60.80**	100	43919	**12.97**	15	48.37	+0.15	+2.00	+3.61
A3	45, RD	τ_{max}	106531	**60.80**	100	44000	**12.99**	17	43.69	+0.11	+1.53	+3.3
A4	45, SB	τ_{max}	106531	**60.80**	100	44316	**13.08**	17	42.92	+0.12	+1.48	+3.3
A5	40, GR	τ_{max}	106531	**60.80**	100	45106	**13.34**	18	40.00	+0.12	+1.49	+3.07
A6	20, RD	τ_{min}	106531	**60.80**	100	64157	**21.65**	23	19.90	-0.01	+0.41	+1.61
A7	30, SB	τ_{min}	106531	**60.80**	100	51313	**15.69**	18	29.88	+0.03	+0.92	+2.38
A8	25, GR	τ_{min}	106531	**60.80**	100	56760	**18.07**	15	25.00	+0.00	+0.82	+1.99
O1	100, RD	τ_{max}	154379	**103.82**	100	62362	**23.66**	22	49.94	+0.32	+1.67	+2.33
O2	100, RD	τ_{min}	154379	**103.82**	100	62389	**23.66**	26	48.76	+0.31	+1.77	+2.31
O3	35, RD	τ_{max}	154379	**103.82**	100	63235	**23.82**	37	33.33	+0.28	+1.59	+1.89
O4	42, SB	τ_{max}	154379	**103.82**	100	63484	**23.83**	35	34.00	+0.28	+1.52	+1.91
O5	40, GR	τ_{max}	154379	**103.82**	100	63095	**23.77**	34	39.99	+0.29	+1.73	+2.08
O6	80, RD	τ_{min}	154379	**103.82**	100	62473	**59.53**	36	34.77	+0.29	+1.58	+1.98
L1	100, RD	τ_{max}	54807	**47.49**	100	29776	**15.73**	31	50.67	+0.42	+1.58	+1.56
L2	100, RD	τ_{min}	54807	**47.49**	100	29794	**15.73**	32	49.86	+0.41	+1.56	+1.53
L3	40, RD	τ_{max}	54807	**47.49**	100	29879	**15.74**	36	37.93	+0.32	+1.32	+1.17
L4	40, SB	τ_{max}	54807	**47.49**	100	30912	**15.94**	35	35.31	+0.3	+0.54	+1.12
L5	40, GR	τ_{max}	54807	**47.49**	100	30818	**15.92**	34	39.99	+0.34	+1.30	+1.27
L6	30, RD	τ_{min}	54807	**47.49**	100	31398	**16.56**	37	29.85	+0.25	+0.22	+0.96
N1	100, RD	τ_{max}	115074	**115.19**	100	56070	**38.39**	21	49.51	+0.03	+1.49	+3.18
N2	100, RD	τ_{min}	115074	**115.19**	100	56076	**39.39**	24	49.95	+0.03	+1.39	+3.16
N3	40, RD	τ_{max}	115074	**115.19**	100	56241	**38.41**	26	36.77	+0.01	+0.86	+2.63
N4	40, SB	τ_{max}	115074	**115.19**	100	59450	**39.11**	28	31.67	-0.02	+0.81	+2.31
N5	45, GR	τ_{max}	115074	**115.19**	100	57471	**38.69**	28	44.21	+0.01	+1.49	+2.89
N6	35, RD	τ_{min}	115074	**115.19**	100	59863	**40.74**	31	34.86	+0.00	+0.84	+2.45

Recall from Section IV.C.1 that for each row, only one orientation (defined by k_i) of the cells satisfies $\tau = \tau_{min}$. As a result, it is likely that the ILP model can become infeasible if $\tau = \tau_{min}$ and F_G is low. Due to this, we find that Experiments 7–8 described above render the ILP model infeasible for some benchmarks (hence, they are omitted from table II). However, for all other cases, there is a considerable reduction in the number of para-FET hotspots and global para-FET susceptibility. The relative reduction in global susceptibility G, defined in Section IV.A, is higher because strong para-FETs (which we preferentially mitigate) have a higher contribution to G. Para-FETs in the linear/saturation range can short a node to VDD (VSS) leads to a conditional stuck-at-1 (stuck-at-0) fault. We insert para-FETs between 100 random cell pairs and observe that ParaMitE reduces the number of conditional stuck-at faults by up to 85% (A1 in Table II). This optimization comes at negligible cost in terms of total power, critical path delay, and wire length. Comparing Experiments 1 and 2, we observe that the performance is similar irrespective of τ. From Experiments 3 and 4, we find that for the same F_G, susceptibility-based budgeting offers similar performance as row-density budgeting with a lower number of cells flipped. While the greedy budgeting policy (Experiment 5) efficiently utilizes the entire flip budget, the reduction in global susceptibility for the same flip count is lower compared to the other two budgeting policies. Experiments 6–8 show that across all budgeting policies, if we limit F_G, the performance with $\tau = \tau_{min}$ is worse compared to when $\tau = \tau_{max}$. This is because several cell-flip solutions are rendered infeasible under strict τ and F_G constraints.

Interestingly, ParaMitE-induced cell flips reduce the impact of the unmitigated para-FETs by reducing the usage of gate metal during re-routing. After executing ParaMitE with $\tau = \tau_{max}$, SB budgeting policy, and $F_G = F_G^*$, the percentage reductions in the wire length of gate-metal routing are 4.3%, 3.1%, 4.2%, and 1.3% for AES, LDPC, Nova, and OpenPiton, respectively. The gate-metal routing is higher before ParaMitE because the PnR tool has placed cells in orientations such that gate-metal routing can connect nearby cells for wire length minimization. Flipping the cells reduces gate-metal routing by introducing blockages. As para-FETs driven by the gate metal are stronger than those driven by other BEOL metal layers, reduction in the gate-metal routing post-ParaMitE decreases the impact of the unmitigated hotspots.

VI. Conclusion

We have presented ParaMitE, an optimization method to reduce the number of potential para-FETs in a CNFET layout. Our method ensures that, even in the presence of unetched CNT aggregates in the CNFET active layer, up to 60% of the para-FETs formed will always be tuned-off. We preferentially mitigate strong para-FETs that short two adjacent cells placed closely. Simulation results, obtained using a CNFET PDK from a commercial foundry, show that up to 90% of these strong para-FETs (which lead to catastrophic stuck-at faults) can be mitigated. ParaMitE includes multiple parameters (e.g., τ, F_G, and budgeting policy) that can be tuned based on the CNFET layout. We discuss how these parameters can be tuned based on the characteristics of the design. The low CPU run time ensures that such iterative tuning is feasible for large designs. In addition, a hybrid flow (where parameters can vary across rows and different ParaMitE settings are considered) can also be explored. As the cells are flipped *in situ*, the impact on the total power, critical path delay, and the wire length are negligible. ParaMitE can be used to identify the locations of para-FET hotspots in a layout and can therefore also be used to provide effective feedback to the foundry for yield ramp-up.

Authorized licensed use limited to: Georgia Institute of Technology. Downloaded on March 12,2022 at 13:12:22 UTC from IEEE Xplore. Restrictions apply.

REFERENCES

[1] M. M. Shulaker, G. Hills, R. S. Park, R. T. Howe, K. Saraswat, H.-S. P. Wong, and S. Mitra, “Three-dimensional integration of nanotechnologies for computing and data storage on a single chip,” *Nature*, vol. 547, pp. 74–78, 2017.

[2] G. S. Tulevski, A. D. Franklin, D. Frank, J. M. Lobez, Q. Cao, H. Park, A. Afzali, S.-J. Han, J. B. Hannon, and W. Haensch, “Toward high-performance digital logic technology with carbon nanotubes,” *ACS Nano*, vol. 8, pp. 8730–8745, 2014.

[3] S. Eachempati, A. Nieuwoudt, A. Gayasen, N. Vijaykrishnan, and Y. Massoud, “Assessing carbon nanotube bundle interconnect for future FPGA architectures,” in *2007 Design, Automation & Test in Europe Conference & Exhibition*. IEEE, 2007, pp. 1–6.

[4] F. Chen, A. Joshi, V. Stojanović, and A. Chandrakasan, “Scaling and evaluation of carbon nanotube interconnects for VLSI applications,” in *Proceedings of the 2nd international conference on Nano-Networks*, 2007, pp. 1–8.

[5] C. Qiu, Z. Zhang, and L.-M. Peng, “Scaling carbon nanotube cmos fets towards quantum limit,” in *2017 IEEE International Electron Devices Meeting (IEDM)*, 2017, pp. 5–5.

[6] S. Rai, S. Srinivasa, P. Cadareanu, X. Yin, X. S. Hu, P.-E. Gaillardon, V. Narayanan, and A. Kumar, “Emerging reconfigurable nanotechnologies: Can they support future electronics?” in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2018.

[7] A. Lin, N. Patil, H. Wei, S. Mitra, and H.-S. P. Wong, “A metallic-CNT-tolerant carbon nanotube technology using asymmetrically-correlated CNTs (ACCNT),” in *Symposium on VLSI Technology*, 2009.

[8] N. Patil, J. Deng, H.-S. Wong, and S. Mitra, “Automated design of misaligned-carbon-nanotube-immune circuits,” in *Proceedings of the 44th annual Design Automation Conference*, 2007.

[9] S. Banerjee, A. Chaudhuri, A. Ning, and K. Chakrabarty, “Variation-aware delay fault testing for carbon-nanotube fet circuits,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 2, pp. 409–422, 2021.

[10] G. Hills, C. Lau, A. Wright, S. Fuller, M. D. Bishop, T. Srimani, P. Kanhaiya, R. Ho, A. Amer, Y. Stein *et al.*, “Modern microprocessor built from complementary carbon nanotube transistors,” *Nature*, vol. 572, no. 7771, pp. 595–602, 2019.

[11] “Printing nanotube-based p-type thin film transistors with high current density,” *NanoIntegris white paper*. [Online]. Available: https://www.nanointegris.com/

[12] T. Srimani, G. Hills, M. D. Bishop, and M. M. Shulaker, “30-nm contacted gate pitch back-gate carbon nanotube FETs for Sub-3-nm nodes,” *IEEE Transactions on Nanotechnology*, vol. 18, pp. 132–138, 2018.

[13] T. Srimani, G. Hills, M. Bishop, C. Lau, P. Kanhaiya, R. Ho, A. Amer, M. Chao, A. Yu, A. Wright *et al.*, “Heterogeneous Integration of BEOL Logic and Memory in a Commercial Foundry: Multi-Tier Complementary Carbon Nanotube Logic and Resistive RAM at a 130 nm node,” in *2020 IEEE Symposium on VLSI Technology*, 2020, pp. 1–2.

[14] M. D. Bishop, G. Hills, T. Srimani, C. Lau, D. Murphy, S. Fuller, J. Humes, A. Ratkovich, M. Nelson, and M. M. Shulaker, “Fabrication of carbon nanotube field-effect transistors in commercial silicon manufacturing facilities,” *Nature Electronics*, vol. 3, no. 8, pp. 492–501, 2020.

[15] R. Thomas, K. Hembram, B. M. Kumar, and G. M. Rao, “High density oxidative plasma unzipping of multiwall carbon nanotubes,” *RSC advances*, vol. 7, no. 76, pp. 48 268–48 274, 2017.

[16] S. Ghosh and K. Roy, “Parameter variation tolerance and error resiliency: New design paradigm for the nanoscale era,” *Proceedings of the IEEE*, vol. 98, no. 10, pp. 1718–1751, 2010.

[17] S. Banerjee, A. Chaudhuri, and K. Chakrabarty, “Analysis of the impact of process variations and manufacturing defects on the performance of carbon-nanotube fets,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 6, pp. 1513–1526, 2020.

[18] D. Z. Pan, B. Yu, and J.-R. Gao, “Design for manufacturing with emerging nanolithography,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 10, pp. 1453–1472, 2013.

[19] K. Cheng, S.-Z. Yao, and T. Hu, “The orientation of modules based on graph decomposition,” *IEEE Computer Architecture Letters*, vol. 40, no. 06, pp. 774–780, 1991.

[20] M. R. Garey and D. S. Johnson, *Computers and intractability. A guide to the theory of NP-completeness*. W.H. Freeman, San Francisco, 1979.

[21] Y. Zhang, “Handling the complexity of routing problem in modern VLSI design,” *Digital Repository, Iowa State University*, 2011.

[22] K. C. Kiwiel, T. Larsson, and P. O. Lindberg, “Lagrangian relaxation via ballstep subgradient methods,” *Mathematics of Operations Research*, vol. 32, no. 3, pp. 669–686, 2007.

[23] C. Yu and Z. Zhang, “Painting on placement: Forecasting routing congestion using conditional generative adversarial nets,” in *Proceedings of the 56th Annual Design Automation Conference 2019*, 2019, pp. 1–6.

Authorized licensed use limited to: Georgia Institute of Technology. Downloaded on March 12,2022 at 13:12:22 UTC from IEEE Xplore. Restrictions apply.