Exploring Physical Synthesis for Circuits based on Emerging Reconfigurable Nanotechnologies

Andreas Krinke*, Shubham Rai†, Akash Kumar† and Jens Lienig*
*Institute of Electromechanical and Electronic Design (IFTE), †Chair For Processor Design (CfAED)
Technische Universität Dresden, Dresden, Germany
Email: {firstname.lastname}@tu-dresden.de

***Abstract*—Recently proposed ambipolar nanotechnologies allow the development of reconfigurable circuits with low area and power overheads as compared to the conventional CMOS technology. However, using a conventional physical synthesis flow for circuits that include gates based on reconfigurable FETs (RFETs) leads to sub-optimal results. This is due to the fact that the physical synthesis flow for circuits based on RFETs has to cater to the additional gate terminal per RFET transistors. In the present work, we explore three important verticals that lead to an optimized physical synthesis flow for RFET-based circuits with circuit-level reconfigurability: (1) designing optimized layouts of reconfigurable gates, (2) utilize special driver cells to drive the reconfigurable portions of a circuit, and (3) optimized placement of these reconfigurable parts in separate power domains. Experimental evaluations over EPFL benchmarks using our proposed approach show a reduction in chip area of up to 17.5 % when compared to conventional flows.**

I. Introduction

With the downscaling of CMOS transistors below 15 nm, the gains in performance are not any longer proportional to the overall cost involved [1]. Hence, several emerging nanotechnologies with exciting properties such as memristors [2], spin-based devices [3], [4] and reconfigurable FETs [5]–[8] are being explored in order to favorably offset this performance-to-cost ratio.

Out of these, reconfigurable FETs (RFETs) belong to a specific class of emerging nanotechnologies which demonstrate ambipolar electrical properties at the device level. Ambipolarity is a phenomenon by which transistors demonstrate both p- and n-type conductivity on application of external bias. Ambipolarity is a virtue of device physical properties at technology nodes below 40 nm and is suppressed in a typical CMOS manufacturing process [9]. However, the same ambipolarity can be enhanced using special process techniques [10] to devise technologies which can show extended logical functionality. This extended logical functionality manifests itself as device-level reconfigurability in RFETs, which can be used to build gates and circuits which offer higher functionality per computation unit [10], [11].

Most of the research in the domain of reconfigurable nanotechnologies has been primarily focused on four main levels. First, at the transistor level, where various materials like carbon [8], graphene [12], silicon [5], [6], germanium [13] and even 2D-materials like graphene p-n junction [7] and WSe_2 [14] have been explored, showcasing electrical symmetry for both p and n-type behavior. Second, at the logic optimization [15], [16] and technology mapping levels [17], [18], where modifications to existing EDA tools have been proposed primarily at the logic synthesis level. Third, at the circuit level where efficient manually designed circuit have been proposed [11], [19], [20]. Lastly, at the fourth level, where various applications have been explored using these technologies [21], [22]. Among applications, device-level reconfigurability is one of the most promising solutions for enabling hardware security [23], [24].

However, a major research aspect which has been missing until now is the support of these works from the physical synthesis point of view, i.e., their compatibility with existing synthesis flows. Earlier works like [25], [26] focused on physical synthesis for RFETs on a configurable sea-of-tiles like fabric. Similarly, physical synthesis for a class of RFETs was proposed in [27] where the authors presented an open source flow based on *lef* and *liberty* files with a benchmark-level evaluation.

In this work, we primarily focus on optimizing area by exploring ways to improve the physical synthesis for RFET-based circuits through creative use of multiple *power domains* and *power shut-off* (PSO). To the best of our knowledge, there has been no prior work on the placement and routing of reconfigurable logic gates based on these emerging technologies, which is a prerequisite for reconfigurable logic circuits. The major contributions are:

- We propose new area-optimized layouts for reconfigurable logic gates based on RFETs.
- We propose two approaches for placement and routing of RFET-based circuits. We use driver cells to generate the P and $\overline{P}$ signals required by the reconfigurable logic gates. Such an approach leads to localization of reconfigurable components of the circuit which can find large applications in hardware security [28]
- Since RFETs are used for reconfigurable logic gates and hence reconfigurable circuits [11], we investigate how the ratio between reconfigurable and conventional logic gates affects chip area, wirelength, and power consumption.

We evaluate our two approaches using the EPFL benchmark suite [29]. Compared to the state of the art as proposed in [27],

The first two authors contributed equally to this work.

Please cite as:

A. Krinke, S. Rai, A. Kumar, J. Lienig, "Exploring Physical Synthesis for Circuits based on Emerging Reconfigurable Nanotechnologies," in Proc. of the Int'l Conf. on Computer-Aided Design (ICCAD), Nov. 2021, Munich, Germany, pp. 1–9

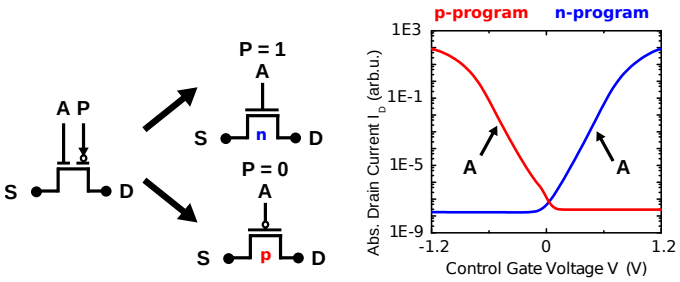


Fig. 1. The figure on the left shows a transistor with two gate terminals – A as CG and P as PG. Changing the potential at P leads to change the electrical characteristics of the device from p to n-type functionality. The graph on the right side shows the electrical symmetry between p and n-type behaviour [10].

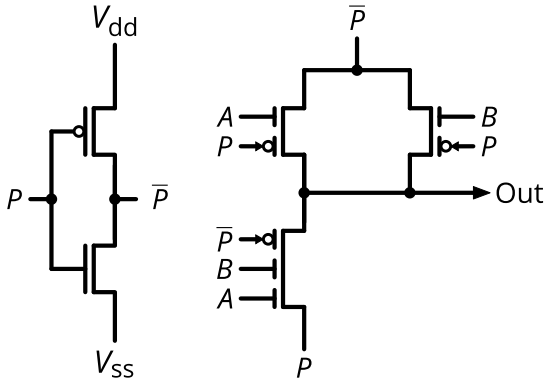


Fig. 2. Schematic of the minority (*MIN*) logic gate proposed in [27] containing an inverter.

we get area improvements of up to 17.5 %, and reduce the static power consumption by 4.4 % on average.

The remainder of the paper is organized as follows. Section II gives details about the motivation of the work and lays the fundamental concepts and background relevant for this work. Section III introduces the concept of driver cells and discusses the overall physical design flow. This is followed by Section IV which discusses our experiments and put forth the results and analysis for our approach. Section V summarizes this paper and presents future works.

II. Background and Motivation

A. Ambipolar Nanotechnologies

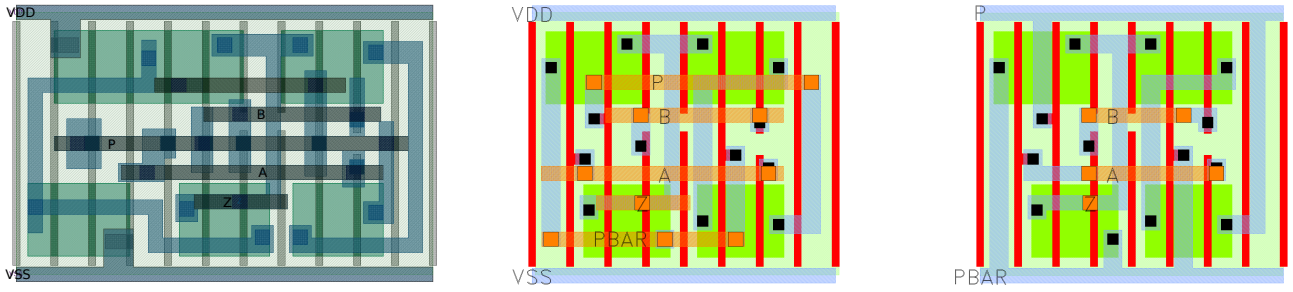
Transistors or nanodevices which can combine the functionality of both p-type and n-type devices in a single device are termed as either *ambipolar devices* [6] or *reconfigurable (field-effect) transistors (RFETs)* [10]. Several devices based on materials like silicon, germanium, carbon, or graphene have been demonstrated to exhibit ambipolar behavior. The electrical symmetry between p- and n-type behavior becomes visible in their *I-V* transfer curves as shown in Fig. 1. Reconfigurable FETs generally come with one or more additional contact terminals which act as control knobs for changing the functionality. In Schottky-junction-based nanowire devices, such as silicon or germanium nanowires, there are two types of gate terminals—the program gate (PG) and the control gate (CG) [30]. The program gate controls the direction of the flow of the charge carriers and hence is responsible for reconfiguring the functionality. The control gate, on the other hand, is analogous to the gate terminal in traditional CMOS transistors, since it controls the flow of charge carriers through the channel.

From a geometry perspective, 1D devices like silicon [5], [6] or germanium [13] nanowires have shown excellent reconfigurable properties. These are enabled by Schottky-junctions between metal-silicon structures. The conductivity is then controlled by two or more independent gate terminals. Just like CMOS transistors, RFETs are also voltage-driven devices. RFETs follow a similar top-down manufacturing process as in the case of CMOS and hence are ahead of other emerging nanotechnologies (such as spin-based devices) in terms of commercial adoption [30], [31].

Similarly, several planar devices (some literature refers them as 2D devices as well) like graphene p-n junctions [7] and recently proposed 2D-transition metal dichalcogenide (TMD) based on materials like $MoTe_2$ [32] and WSe_2 [14] have also been demonstrated to exhibit reconfigurable properties at the device-level.

B. Motivation

Due to the inherent ambipolarity offered by these emerging reconfigurable nanotechnologies, transistors and hence logic



(a) (b) (c)

Fig. 3. (a) Layout of the minority (*MIN*) logic gate proposed in [27] containing an inverter. (b) Our newly proposed layout of a *MIN* gate with P, $\overline{P}$ (PBAR) as separate input signals. (c) Our improved *MIN* gate layout for the PSO approach.

gates demonstrate runtime-reconfigurability by applying different bias voltages to the program gate of a single or a group of transistors [11]. This runtime-reconfigurability can be exploited for developing polymorphic logic gates [33] at low area, power and delay overheads. Several such reconfigurable logic gates such as *AOI-OAI21* and even *NAND3-NOR3* have been demonstrated in prior works [10], [11]. Such polymorphism allows to achieve efficient solutions for hardware security, particularly for logic locking schemes [4], [34], [35].

However, these earlier works have overlooked the physical design flow for circuits based on RFETs. Hence, the impact of security measures over routing and placement have been missing which leads to a more abstract overheads in terms of area and power. The present work looks at physical synthesis to support such logic-locking-based hardware security schemes. We explore various approaches in physical synthesis flow for RFETs-based circuits and discuss concrete area and power numbers by carrying out a benchmark-level evaluation. We also discuss how the variability of reconfigurable logic further affects the area and other parameters of the circuit. We refrain from security analysis as these evaluations using logic-level modelling have been carried over RFETs-based circuits in previous works such as [27], [34], [36]. These works clearly show that practical security can easily be achieved against seminal SAT-based attacks [37] using RFET-based circuits[1]. Hence, for the sake of reader's understandability, we take a trivial example of logic-locking in which the starting RFET-based circuit is only mapped with *NAND* and *NOR* logic gates. We then replace some portions of these gates with *MINORITY* gate (a polymorphic gate which can be configured either as *NAND* or *NOR*) and carry out physical synthesis to demonstrate our approach and findings.

C. Previous Works

In terms of circuit design flow, most of the earlier works targeted either at logic optimization [15], [17], [18] or at the level of manually designed circuit elements [19], [39], [40]. While they showed great improvements in the overall functional capabilities of RFETs, an important consideration of physical synthesis for RFET based circuits has been overlooked. For RFETs, where a single device has more contact terminals as compared to traditional CMOS, placement and routing are more challenging for conventional approaches. Although RFETs lose to CMOS in terms of number of contacts per transistor, they make up for this with their higher functional expression, since the overall number of transistors per circuit is greatly reduced. This has been demonstrated particularly in [11], [20].

There have been relatively fewer attempts to target physical synthesis flows for RFETs-based circuits [26], [27]. The authors in [26] proposed a technology-independent sea-of-tiles fabric where individual logic gates can be mapped easily.

[1]For more information over logic-locking and SAT-based attacks, readers are requested to refer [38]

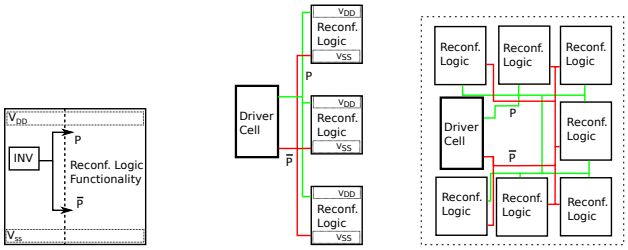


Fig. 4. Approaches for physical synthesis of RFETs. (a) Layout of reconfigurable logic gate proposed in [27]. P and $\overline{P}$ signals are driven by inverting logic embedded within the logic gate. (b) Out native approach which removes the inverting logic and uses separate driver cells for generating P and $\overline{P}$. (c) Our PSO-based approach that groups reconfigurable logic gates in power domains which are driven by common P and $\overline{P}$ signals.

However, such a sea-of-tiles fabric is incompatible with conventional physical synthesis flows. Hence, it is required to design a complete EDA flow comprising both logic and physical synthesis for RFET-based circuits [27]. The authors proposed layouts for logic gates and also for reconfigurable logic gates such as minority (*MIN*), presenting an end-to-end open-source flow. However, the layouts they discussed were pessimistic because they assumed for every reconfigurable logic gate an inverter within the cell layout boundary. This layout design had a negative impact on area, delay and power. In the present work, we start with the open-source flow [27] and propose new layouts and placement approaches to optimize physical synthesis for RFET-based circuits.

III. Exploring Physical Synthesis

In [27], the authors proposed layouts of reconfigurable logic cells. These gates contain an inverter (within the gate boundary), so that both P and $\overline{P}$ are available for reconfiguration. The *MIN* gate in Fig. 2 (schematic) and Fig. 3a (layout) is such a gate [27], in which the inverter is located on the left side, connected to V_{dd} and V_{ss}. A diagrammatic representation is shown in Fig. 4a. In this case, M1 metal wires were used for V_{dd} and V_{ss}, while the P and $\overline{P}$ signals were routed together with all other signals using polysilicon, M1, and M2. Because V_{dd} and V_{ss} are only needed for the operation of the internal inverter, their wiring wastes valuable routing resource that cannot be used to connect P, $\overline{P}$, and all other signals. This also leads to an increased pin density and an area overhead for each reconfigurable logic gate in the circuit. In our work, we mitigate these issues by removing the inverter which leads to a smaller cell area, decreased pin density, and more available routing resources for the *P&R* tool.

A. Concept of Driver Cells

In order to enable reconfigurable functionality using RFET-based logic gates, an inverting logic is imperative to generate $\overline{P}$ from P [11]. However, we notice that within a particular circuit, multiple reconfigurable logic gates share the same value for P.

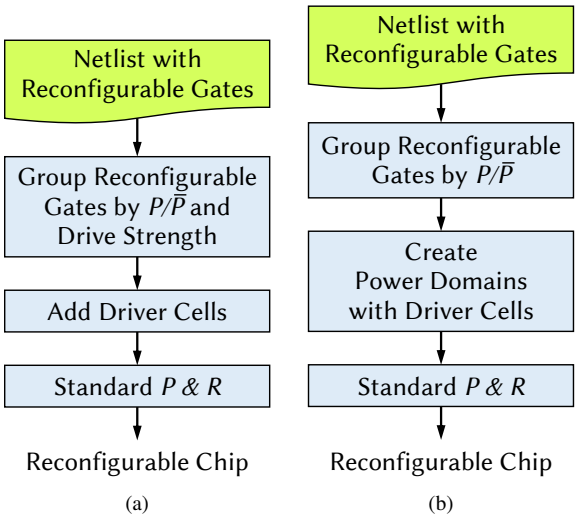


Fig. 5. Our design flows for (a) the native approach and (b) the island-based approach.

Hence, for each group of reconfigurable logic gates sharing the same P and $\overline{P}$, we instantiate a *driver cell* which particularly powers these two signals. These driver cells comprise larger inverters with higher fan-out, which are usually included in the process design kits (PDKs). This allows us to introduce new layouts of reconfigurable logic gates, as shown in Figs. 3b and 3c, which do not contain an inverter. They need to be connected to P and $\overline{P}$ of a driver cell instead. In order to perform placement and routing for these circuits, we propose a native as well as an island-based approach. Both are described in the following sections.

B. Native Approach

In this approach, reconfigurable logic cells are stripped of their internal inverter, as explained above. As shown in Fig. 3b for the *MIN* gate, this obviously leads to a smaller gate area. which improves total area and power consumption. The V_{dd} and V_{ss} contacts at the top and bottom are not changed so that the gate can be placed on the default power rails like any other standard cell. However, in these cells, the V_{dd} and V_{ss} contacts are not really needed. Instead, the gates are powered by a driver cell using P and $\overline{P}$. This leads to a waste of M1 routing resources, which increases pin density and limits the possible area improvements.

The resulting design flow is shown in Fig. 5a. It starts with a Verilog netlist that includes reconfigurable gates. As shown in Fig. 4b, a single driver cell powers multiple reconfigurable logic gates using P and $\overline{P}$. Depending on the available drive strengths of the $P/\overline{P}$ drivers, we create groups with the correct number of reconfigurable gates that share the same $P/\overline{P}$. For each group, a driver is added to the netlist and connected to the individual reconfigurable gates. The driver's input signal P can then be connected to the corresponding pin or signal. Since the new reconfigurable logic gates can be placed and routed like any other standard cell, we can now perform standard *P&R* to generate the layout.

It can be argued that from hardware security point of view, sharing of signals can compromise the robustness of a circuit against security attacks. However, in any logic-locked circuit, locking signals are limited by the size of tamper-proof memory [38] and hence, all of them cannot be possibly made as primary inputs. This promotion of signals from an intermediate signal to primary inputs also adds to routing and area overheads. We leave this decision on the hardware designer to explore the amount of sharing, a particular design can afford for maintaining an acceptable level of security.

C. Island-based Approach

The biggest limitation of the native approach are the unused V_{dd} and V_{ss} contacts on M1, which only serve the compatibility of the cells with the default power rails. To circumvent this issue, we removed the connections to V_{dd} and V_{ss} and reused their metal contacts for P and $\overline{P}$. The improved cell layout is shown in Fig. 3c. This approach greatly reduces pin density on M2 in areas with many reconfigurable logic cells. Due to the simpler routing, the total area of the chip and the wirelength are also reduced considerably.

One of the limitations for this approach is that these new and improved cells can no longer be placed on the V_{dd} and V_{ss} power rails. To still carry out the placement and routing of RFET-based circuits using these improved cell layouts, we apply an island-based approach using multiple power domains and *Power-Shut-Off* (PSO) mode, which is supported by various *P&R* tools for low power design. In the island-based placement, groups of reconfigurable logic gates that share P and $\overline{P}$ will be placed in their own rectangular power domain. As shown in Fig. 4c, we masquerade the driver cells as power switches that are automatically placed in the power domain and automatically routed by the *P&R* tool using PSO mode. Since P and $\overline{P}$ use the power rails in the power domain, no additional resources are needed for routing. By using the switching signal which was originally intended to switch the power domain on and off, we can control the driver cells and reverse the polarity of P and $\overline{P}$. This makes it possible to reconfigure all logic gates in the power domain simultaneously, depending on the functionality required by the circuit.

Our design flow for this island-based approach is shown in Fig. 5b. It starts with the same netlist as the native approach from the previous section. Again, we create groups of reconfigurable gates that are connected to the same $P/\overline{P}$ signals. This time, however, there is no limit for the size of these groups. For each group, a separate power domain is set up with the appropriate number of driver cells. As shown in Fig. 6, these driver cells are automatically placed in a grid and drive $P/\overline{P}$ using the M1 power rails (cf. Fig. 3c). Finally, we can perform standard *P&R* to generate the layout.

IV. Experiments and Discussion

A. Experimental Setup

In this section, we present our experiments with the two new approaches for optimized physical synthesis for emerging reconfigurable nanotechnologies. For our experiments, we have

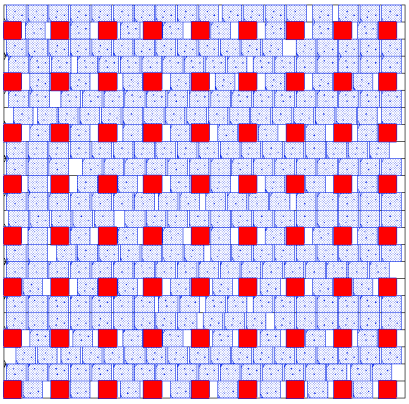


Fig. 6. Power domain for reconfigurable gates that share the same $P/\overline{P}$. Driver cells are highlighted in red. In this example, we added one driver cell for five reconfigurable gates.

used the silicon nanowire based RFET model as proposed in [27]. The *P&R* was carried out using Cadence® Encounter®.

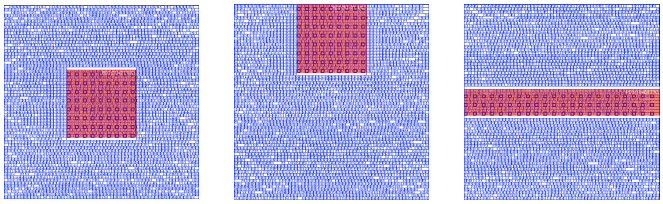
We applied our two methods to the 18 EPFL benchmark circuits [29]. First we map these circuits using the ABC logic synthesis tool [41] and a target library containing only *NAND* and *NOR* gates. To create reconfigurable circuits, we then replace a part of these *NAND* and *NOR* gates with reconfigurable *MIN* gates (cf. Fig. 3). These *MIN* gates represent the reconfigurable logic gates used for hardware security applications [35]. While the baseline netlist uses the *MIN* gate layouts from [27] shown in Fig. 3a, the netlists of both the native approach and the island-based approach use the new layouts shown in Figs. 3b and 3c, respectively.

The *NAND* and *NOR* gates to be replaced are selected evenly across the entire netlist. For this purpose, we iterate over all candidate gates and select a gate if the current replacement ratio is below the desired replacement ratio. Since reconfigurable gates are placed in separate power domains during the island-based approach, we improve this selection by performing an additional optimization. For this we apply the Fiduccia-Mattheyses (FM) algorithm [42] to reduce the number of cut nets between the reconfigurable logic gates and the rest of the circuit. Starting with the initial partitioning, we repeatedly perform two steps: (1) move a gate *out* of the power domain, and (2) move another gate *into* the power domain. By combining the two steps, the number of gates in the power domain and, therefore, the replacement ratio remain constant. The gates are selected in the order of their gain, i.e. the more a gate reduces the number of cut nets, the sooner it is moved. Only if there is an improvement, the two moves are actually executed. After all gates in the power domain have been considered and as long as the number of cut nets has been reduced, the whole process starts again.

The entire process of

- partitioning and transforming the netlist,
- template-based creation of the TCL scripts and CPF files to control Cadence Encounter,
- running *P&R*,
- creating the layout images (cf. Fig. 8), and
- extracting the relevant parameters from the log files

is automated using Python.



(a) (b) (c)

Fig. 7. Power domain configurations that we investigated in our experiments: (a) square in the center, (b) square in the top center, and (c) rectangle covering the whole width in the center.

B. Utilization Factor

When placing and routing a netlist, adjusting the utilization factor has the greatest influence on the chances of success and the quality of the layout. This factor defines the ratio between used and unused area during placement. For example, a utilization factor of 0.9 means that 90 % of the core area is occupied by macros and standard cells while 10 % of the area remains empty. The smaller this factor is, the larger the layout will be, resulting in lower cell and pin densities. As a result, routing becomes possible in the first place, or at least it is simplified.

In our experiments, we start with a core utilization of 0.8 and a power domain utilization of 0.9. We manually reduced these values for each benchmark depending on the location of DRC errors until the resulting layout had no design rule violations.

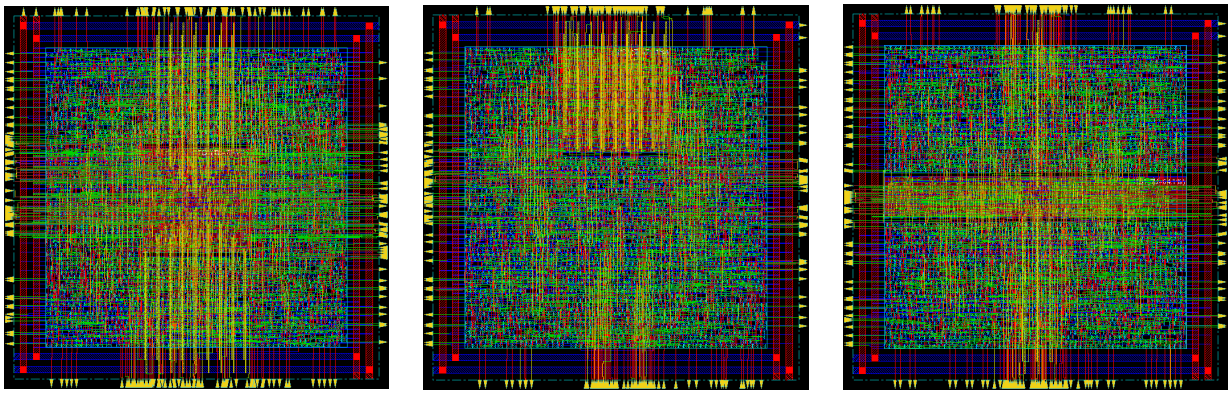
C. Placement of the Island on the Chip

For the island-based approach, the placement of the power domain in the core is an important consideration. Figure 7 shows the three variants that we have examined experimentally. We conclude that there is not a single configuration that is the best for every test case. Instead, the choice hugely depends on the circuit structure, the amount of reconfigurable logic in a circuit, and whether we want to minimize the area or the wirelength.

As shown in Fig. 7, there is an empty row above and below of the power domain. This is necessary because of the different power rail voltages in the power domain and outside. As a result, configuration (c) in Fig. 7 has a larger unused area than configuration (a). The configuration in (b) has the smallest unused area due to the placement of the power domain at the edge of the circuit. This difference can have a significant influence on the achievable area, especially with smaller circuits.

D. Results

In our experiments we apply both approaches to the circuits of the EPFL benchmark suite [29]. Example layouts for the *bar* benchmark using the island-based approach are shown in Fig. 8. Figure 9 shows the results of our experiment for



(a) (b) (c)

Fig. 8. Layouts of the EPFL *bar* benchmark with 10 % reconfigurable logic using the island-based approach. The three layouts correspond to the floorplans in Fig. 7.

a replacement ratio of 0.1. Both approaches described in Section III (native and island-based approach) are shown in comparison with the baseline which is the vanilla version of physical synthesis using the RFET model from [27]. As we can see in Fig. 9a, the core area is reduced by 1.5 % on average (2.3 % when ignoring the outlier benchmark *div*) for the native approach and by 0.2 % for the island-based approach. These values are quite low for two reasons. First, we are loosing a large part of the area advantage—which we got by removing the inverter from the reconfigurable logic gates—by adding one driver cell for each group of five[2] reconfigurable logic gates. This shows the great impact of the number of gates a driver cell can drive. Second, the island-based approach has a high overhead due to the empty rows above and below, which makes the area reduction even worse on average compared to the native approach.

However, as shown in Fig. 9b, the island-based approach has a much smaller impact on wirelength. This is caused by the much higher pin density when using the native approach, because each reconfigurable logic gate has two additional pins, $P/\overline{P}$. With the island-based approach, these pins are connected using the power rails, which reduces the pin density and, thereby, wirelength. This is due to the high pin density of the standard cell layout as the original layout of the logic gates as proposed in [27] are very CMOS-like, hence ignoring the structural differences with the RFETs. This does not allow optimal routing for metal layers M2 within the island. As a result, the P and $\overline{P}$ routing is done either through metal layers M3 or M4 in order to make the physical synthesis DRC error free. This can however be mitigated with better standard cell designs using a more advanced PDK. Presently, the available PDK is based on CMOS and hence is not optimized for RFETs. Specially RFET-based PDK can alleviate these issues and can also enable further area reduction with better standard cell designs.

The static power consumption, as reported by Cadence Encounter, is reduced in both approaches. In contrast, the dynamic power consumption is increased for most of the benchmarks. This increase is far worse for the native approach, where it correlates with the increased wirelength.

When looking at the influence of the power domain configuration discussed in Section IV-C, a square in the top center (Fig. 7b) minimizes the core area for all benchmark circuits. For wirelength, a square power domain at the top or in the center is generally the best (Fig. 7a or Fig. 7b), except for some very large circuits (benchmarks *voter*, *square*, *log2*) where the configuration in Fig. 7c produces the layout with the shortest wirelength.

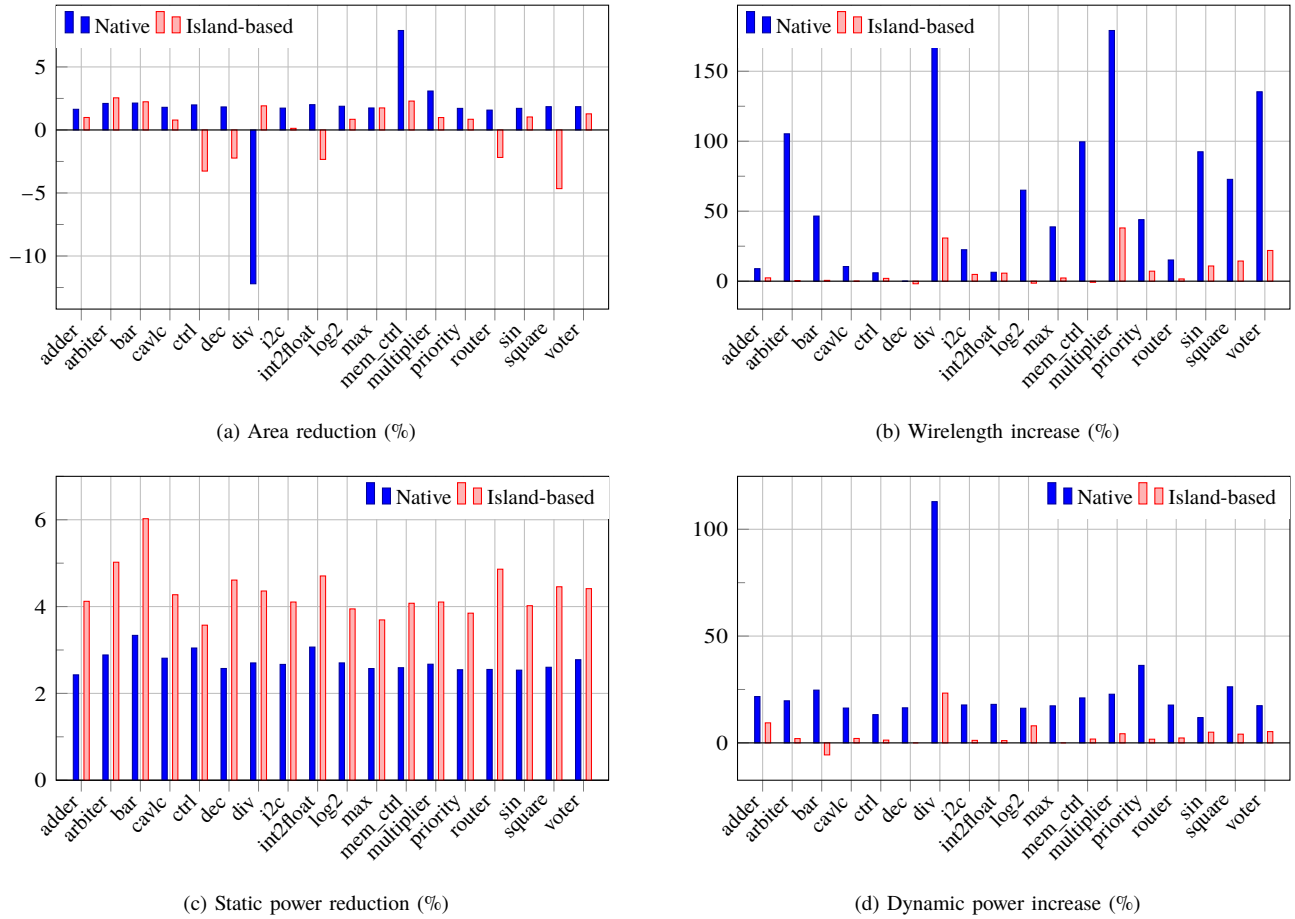
In a second experiment we apply both approaches to the benchmark *bar* using a replacement ratio from 0.1 to 1.0. The results are shown in Fig. 10. As we can see, the island-based approach is far superior for larger replacement ratios. The area overhead decreases with more reconfigurable logic gates and larger power domains. As a result, the area is reduced by up to 17.5 % for a replacement ratio of 0.9. Again, the wirelength increase is much lower for the island-based approach, with a maximum of 13.0 % for a replacement ratio of 0.6. The static power reduction in Fig. 10c is proportional to the replacement ratio. The change in dynamic power consumption varies between a decrease of 13.1 % and an increase of 5.9 %.

For the island-based approach, we get the smallest area using the floorplan in Fig. 7b with the power domain centered at the top. The optimal configuration for the shortest wirelength depends on the replacement ratio. For a replacement ratio of 0.1 or 0.2, the floorplan in Fig. 7b produces the shortest wirelength. For a replacement ratio between 0.3 and 0.5, the centered square in Fig. 7a is the best configuration with regard to wirelength. If more than 50 % of all gates are reconfigurable, the three configurations produce similar results with no clear winner.

E. Discussion

From our exploration study, we can see that there are a lot of parameters which effect the physical synthesis of the circuits based on RFETs. One can see that island-based approach is

[2]We have chosen 5 here as a representative number for our experiments. However, the driver cell can be designed with any drive strength.



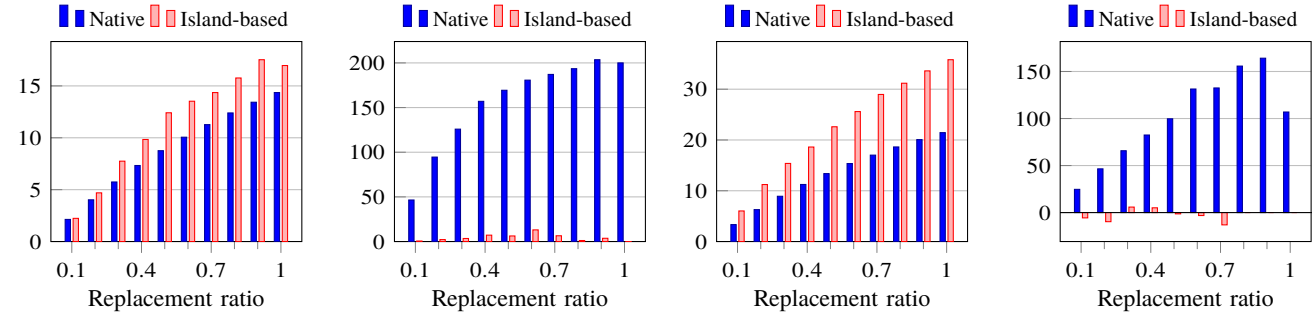
(a) Area reduction (%)

(b) Wirelength increase (%)

(c) Static power reduction (%)

(d) Dynamic power increase (%)

Fig. 9. Comparison of the native approach and the island-based approach with the baseline for a replacement ratio of 0.1, i.e. 10 % of the logic gates are replaced by reconfigurable logic gates. For each EPFL benchmark, the changes in (a) core area, (b) total routed wirelength, (c) static power consumption, and (d) dynamic power consumption from the baseline are shown. For the island-based approach we use the best results of the three power domain configurations in Fig. 7. Please note that in (a) and (c) higher values are better, while in (b) and (d) lower values are desired.



(a) Area reduction (%)

(b) Wirelength increase (%)

(c) Static power reduction (%)

(d) Dynamic power increase (%)

Fig. 10. Changes in (a) core area, (b) total routed wirelength, (c) static power consumption, and (d) dynamic power consumption from the baseline for the EPFL *bar* benchmark with the amount of reconfigurable logic ranging from 10 to 100 %. For the island-based approach we used the best results of the three power domain configurations in Fig. 7. Please note that in (a) and (c) higher values are better, while in (b) and (d) lower values are desired.

more suitable and gives consistent better results in terms of area and wirelength as compared to the native-based approach. Following findings can be noted from our experiments:

- All three approaches—baseline, native, or island-based approach—can be used depending upon the parameter to optimize.
- The factor of replacement ratio which is basically governed by the security assurance has a huge impact in terms of area and wirelength.
- Placement of the power domains depends on factors such as the size of the circuit, connections between the subcircuits and also the layout of the standard cells. Learning-based approaches seem suitable to effectively predict optimized power domain placement. Alternatively, top-level power domain position requirements and gate-level security requirements can be translated into constraints that can be efficiently communicated through top-down and bottom-up propagation within the design hierarchy [43]. These propagated constraints can then support informed configuration of the power domains [44].
- Quality of *liberty* and *lef* files plays an important role in placement and routing. Factors such as pitch between the gate terminals and pin density need to be optimized for shorter total routed wirelength.

V. Summary and Future Works

In this work, we present two new approaches for optimized physical synthesis for emerging reconfigurable nanotechnologies. RFETs offer a new paradigm as compared to CMOS because of their reconfigurable properties where logic gates can be either reconfigurable or static. Such reconfigurable components more often constitute the control segment of the circuit as such features are highly suited for applications in hardware security. Keeping that in mind, we have explored two strategies for physical synthesis with different placement techniques to obtain separate areas for reconfigurable logic gates and static logic gates in the circuit. Using a PSO-inspired island-based approach, we were able to improve on parameters like static power consumption and area.

To further improve physical synthesis for reconfigurable circuits, possibilities for more efficient generation of $P/\overline{P}$ signals should be explored. However, such endeavors require better layouts and specialized PDKs for logic gates based on RFETs. Furthermore, measures like all-around devices and stacking of nanowires [45], [46] will lead to a substantial reduction of device width and accordingly to a reduced cell area of the layouts. This can impact the overall placement of individual standard cells and hence the routing. A higher number of metal layers and a smaller pitch between gate terminals can further relax routing constraints in order to facilitate physical synthesis for RFETs. Similarly, development of libraries for multi-independent gate RFETs (MIGFETs or TIGFETs) offer the potential to improve the overall post-physical synthesis of RFET-based circuits.

Acknowledgments

This research was supported by the German Research Foundation (DFG), project SecuReFET (Project Number: 439891087).

References

[1] M. Y. Vardi, "Moore's law and the sand-heap paradox," *Commun. ACM*, vol. 57, no. 5, May 2014. [Online]. Available: http://doi.acm.org/10.1145/2600347

[2] L. Chua, "Memristor—the missing circuit element," *IEEE Trans. on Circuit Theory*, vol. 18, no. 5, Sep. 1971.

[3] J. Kim, A. Paul, P. A. Crowell, S. J. Koester, S. S. Sapatnekar, J. Wang, and C. H. Kim, "Spin-based computing: Device concepts, current status, and a case study on a high-performance microprocessor," *Proceedings of the IEEE*, vol. 103, no. 1, Jan. 2015.

[4] S. Patnaik, N. Rangarajan, J. Knechtel, O. Sinanoglu, and S. Rakheja, "Advancing hardware security using polymorphic and stochastic spin-hall effect devices," in *DATE*, 2018.

[5] A. Heinzig, S. Slesazeck, F. Kreupl, T. Mikolajick, and W. M. Weber, "Reconfigurable silicon nanowire transistors," *Nano Letters*, vol. 12, no. 1, 2012.

[6] M. De Marchi, D. Sacchetto, S. Frache, J. Zhang, P.-E. Gaillardon, Y. Leblebici, and G. De Micheli, "Polarity control in double-gate, gate-all-around vertically stacked silicon nanowire fets," in *IEDM*, 2012.

[7] S. Tanachutiwat, J. U. Lee, W. Wang, and C. Y. Sung, "Reconfigurable multi-function logic based on graphene p-n junctions," in *DAC*, 2010.

[8] J. Liu, I. O'Connor, D. Navarro, and F. Gaffiot, "Design of a Family of Novel CNTFET-based Dynamically Reconfigurable Logic Gates," in *NEWCAS*, 2007.

[9] M. D. Marchi, D. Sacchetto, J. Zhang, S. Frache, P. E. Gaillardon, Y. Leblebici, and G. D. Micheli, "Top-down fabrication of gate-all-around vertically stacked silicon nanowire fets with controllable polarity," *IEEE Transactions on Nanotechnology*, vol. 13, no. 6, pp. 1029–1038, Nov 2014.

[10] J. Trommer, A. Heinzig, T. Baldauf, S. Slesazeck, T. Mikolajick, and W. M. Weber, "Functionality-enhanced logic gate design enabled by symmetrical reconfigurable silicon nanowire transistors," *TNano*, vol. 14, no. 4, Jul. 2015.

[11] S. Rai, J. Trommer, M. Raitza, T. Mikolajick, W. M. Weber, and A. Kumar, "Designing efficient circuits based on runtime-reconfigurable field-effect transistors," *TVLSI*, vol. 27, no. 3, Mar. 2019.

[12] N. Harada, K. Yagi, S. Sato, and N. Yokoyama, "A polarity-controllable graphene inverter," *APL*, vol. 96, no. 1, 2010.

[13] J. Trommer, A. Heinzig, A. Heinrich, P. Jordan, M. Grube, S. Slesazeck, T. Mikolajick, and W. M. Weber, "Material prospects of reconfigurable transistor (rfets) – from silicon to germanium nanowires," *MRS Proceedings*, vol. 1659, 2014.

[14] G. V. Resta, S. Sutar, Y. Balaji, D. Lin, P. Raghavan, I. Radu, F. Catthoor, A. Thean, P.-E. Gaillardon, and G. de Micheli, "Polarity control in wse2 double-gate transistors," *Sci. Rep.*, vol. 6, Jul. 2016.

[15] L. Amarú, P.-E. Gaillardon, and G. De Micheli, "Majority-inverter graph: A new paradigm for logic optimization," *TCAD*, vol. 35, no. 5, May 2016.

[16] S. Rai, H. Riener, G. D. Micheli, and A. Kumar, "Preserving self-duality during logic synthesis for emerging reconfigurable nanotechnologies," in *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2021.

[17] S. Rai, M. Raitza, and A. Kumar, "Technology mapping flow for emerging reconfigurable silicon nanowire transistors," in *DATE*, 2018.

[18] S. Rai, M. Raitza, S. S. Sahoo, and A. Kumar, "Discern: Distilling standard cells for emerging reconfigurable nanotechnologies," in *2020 Design, Automation Test in Europe Conference Exhibition (DATE)*, March 2020.

[19] P.-E. Gaillardon, L. Amaru, J. Zhang, and G. De Micheli, "Advanced system on a chip design based on controllable-polarity fets," in *DATE*, 2014.

[20] M. Raitza, A. Kumar, M. Völp, D. Walter, J. Trommer, T. Mikolajick, and W. M. Weber, "Exploiting transistor-level reconfiguration to optimize combinational circuits," in *DATE*, 2017.

[21] P. E. Gaillardon, X. Tang, G. Kim, and G. De Micheli, "A novel fpga architecture based on ultrafine grain reconfigurable logic cells," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 10, pp. 2187–2197, oct 2015. [Online]. Available: http://ieeexplore.ieee.org/document/6918535/

[22] A. Chen, X. S. Hu, Y. Jin, M. Niemier, and X. Yin, "Using emerging technologies for hardware security beyond pufs," in *DATE*, 2016.

[23] Y. Bi, X. S. Hu, Y. Jin, M. Niemier, K. Shamsi, and X. Yin, "Enhancing hardware security with emerging transistor technologies," in *Proceedings of the 26th Edition on Great Lakes Symposium on VLSI*, ser. GLSVLSI '16. New York, NY, USA: ACM, 2016, pp. 305–310. [Online]. Available: http://doi.acm.org/10.1145/2902961.2903041

[24] J. Knechtel, "Hardware security for and beyond cmos technology: An overview on fundamentals, applications, and challenges," in *Proceedings of the 2020 International Symposium on Physical Design*, ser. ISPD '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 75–86. [Online]. Available: https://doi.org/10.1145/3372780.3378175

[25] S. Bobba, M. De Marchi, Y. Leblebici, and G. De Micheli, "Physical synthesis onto a sea-of-tiles with double-gate silicon nanowire transistors," in *DAC*, 2012.

[26] S. Bobba and G. De Micheli, "Layout technique for double-gate silicon nanowire fets with an efficient sea-of-tiles architecture," *TVLSI*, vol. 23, no. 10, Oct. 2015.

[27] S. Rai, A. Rupani, D. Walter, M. Raitza, A. Heinzig, T. Baldauf, J. Trommer, C. Mayr, W. M. Weber, and A. Kumar, "A physical synthesis flow for early technology evaluation of silicon nanowire based reconfigurable fets," in *DATE*, 2018.

[28] M. Rostami, F. Koushanfar, and R. Karri, "A primer on hardware security: Models, methods, and metrics," *Proceedings of the IEEE*, vol. 102, no. 8, Aug. 2014.

[29] L. Amarú, P.-E. Gaillardon, and G. De Micheli, "The epfl combinational benchmark suite," in *IWLS*, no. CONF, 2015.

[30] T. Mikolajick, A. Heinzig, J. Trommer, T. Baldauf, and W. M. Weber, "The rfet—a reconfigurable nanowire transistor and its application to novel electronic circuits and systems," *Semicond. Sci. Technol.*, vol. 32, no. 4, Mar. 2017.

[31] M. Simon, A. Heinzig, J. Trommer, T. Baldauf, T. Mikolajick, and W. M. Weber, "Top-down technology for reconfigurable nanowire fets with symmetric on-currents," *IEEE Transactions on Nanotechnology*, vol. 16, no. 5, pp. 812–819, Sept 2017.

[32] S. Nakaharai, M. Yamamoto, K. Ueno, Y.-F. Lin, S.-L. Li, and K. Tsukagoshi, "Electrostatically reversible polarity of ambipolar α-mote2 transistors," *ACS Nano*, vol. 9, no. 6, May 2015. [Online]. Available: https://doi.org/10.1021/acsnano.5b00736

[33] C. A. Mack, "Fifty years of moore's law," *IEEE Trans. on Semiconductor Manufacturing*, vol. 24, no. 2, May 2011.

[34] Y. Bi, X. S. Hu, Y. Jin, M. Niemier, K. Shamsi, and X. Yin, "Enhancing hardware security with emerging transistor technologies," in *GLSVLSI*, 2016.

[35] A. Rupani, S. Rai, and A. Kumar, "Exploiting emerging reconfigurable technologies for secure devices," in *Euromicro DSD*, 2019.

[36] S. Rai, S. Patnaik, A. Rupani, J. Knechtel, O. Sinanoglu, and A. Kumar, "Security promises and vulnerabilities in emerging reconfigurable nanotechnology-based circuits," *IEEE Transactions on Emerging Topics in Computing*, pp. 1–1, 2020. [Online]. Available: https://doi.org/10.1109/TETC.2020.3039375

[37] P. Subramanyan, S. Ray, and S. Malik, "Evaluating the security of logic encryption algorithms," in *2015 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, May 2015, pp. 137–143.

[38] A. Chakraborty, N. G. Jayasankaran, Y. Liu, J. Rajendran, O. Sinanoglu, A. Srivastava, Y. Xie, M. Yasin, and M. Zuzak, "Keynote: A disquisition on logic locking," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 10, pp. 1952–1972, 2020.

[39] J. Zhang, X. Tang, P.-E. Gaillardon, and G. De Micheli, "Configurable circuits featuring dual-threshold-voltage design with three-independent-gate silicon nanowire fets," *TCAS*, vol. 61, no. 10, Oct. 2014.

[40] L. Amarú, P.-E. Gaillardon, J. Zhang, and G. De Micheli, "Power-gated differential logic style based on double-gate controllable-polarity transistors," *TCAS-II*, vol. 60, no. 10, Oct. 2013.

[41] R. Brayton and A. Mishchenko, *ABC: An Academic Industrial-Strength Verification Tool*. Springer Berlin Heidelberg, 2010.

[42] C. M. Fiduccia and R. M. Mattheyses, "A linear-time heuristic for improving network partitions," in *DAC*, 1982.

[43] A. Krinke, G. Jerke, and J. Lienig, "Constraint Propagation Methods for Robust IC Design," in *Proc. 8th Symp. on Reliability by Design (ZuE)*, 2015, pp. 7–14. [Online]. Available: https://ieeexplore.ieee.org/document/7348509

[44] A. Krinke, L. Lei, and J. Lienig, "Predictive system-level constraint verification and optimization," in *Proc. 9th Symp. on Reliability by Design (ZuE)*, 2017, pp. 40–45. [Online]. Available: https://ieeexplore.ieee.org/document/8084535

[45] H. Mertens, R. Ritzenthaler, A. Chasin, T. Schram, E. Kunnen, A. Hikavyy, L. Å. Ragnarsson, H. Dekkers, T. Hopf, K. Wostyn, K. Devriendt, S. A. Chew, M. S. Kim, Y. Kikuchi, E. Rosseel, G. Mannaert, S. Kubicek, S. Demuynck, A. Dangol, N. Bosman, J. Geypen, P. Carolan, H. Bender, K. Barla, N. Horiguchi, and D. Mocuta, "Vertically stacked gate-all-around si nanowire cmos transistors with dual work function metal gates," in *IEDM*, Dec. 2016, pp. 19.7.1–19.7.4.

[46] T. Ernst, C. Dupre, C. Isheden, E. Bernard, R. Ritzenthaler, V. Maffini-Alvaro, J.-C. Barbé, F. De Crecy, A. Toffoli, C. Vizioz *et al.*, "Novel 3d integration process for highly scalable nano-beam stacked-channels gaa (nbg) finfets with hfo2/tin gate stack," in *2006 International Electron Devices Meeting*. IEEE, 2006, pp. 1–4.