

Technology Mapping for Cryogenic CMOS Circuits

Benjamin Hien¹, Marcel Walter^{1,2}, Victor M. van Santen¹, Florian Klemme³, Shivendra Singh Parihar^{3,4},
Girish Pahwa⁵, Yogesh S. Chauhan⁴, Hussam Amrouch^{1,7}, and Robert Wille^{1,6}

¹Technical University of Munich, Germany, ²University of Bremen, Germany,

³University of Stuttgart, Germany, ⁴IIT Kanpur, India,

⁵University of California, Berkeley, USA, ⁶Software Competence Center Hagenberg GmbH, Austria

⁷Munich Institute of Robotics and Machine Intelligence, Munich, Germany Email: benjamin.hien@tum.de

Abstract—Cryogenic CMOS circuits have garnered significant attention for their potential applications in fields such as quantum computing, magnetic resonance imaging, particle detectors, and space missions. Operating at temperatures below 77 K down to almost absolute zero, these circuits face stringent power constraints due to the limited cooling power available at deep cryogenic temperatures. While cryogenic operation can substantially reduce leakage current and improve transistor efficiency, it is crucial to optimize cryogenic CMOS circuits for minimal static and dynamic power consumption to operate within the cooling constraints.

In this paper, we present a cryogenic-aware technology mapping approach to optimize the power characteristics of cryogenic CMOS circuits. The proposed method takes a technology-independent logic network and a cryogenic standard-cell library as input and produces a technology-mapped gate-level netlist with significantly reduced power consumption. By considering static and dynamic power constraints at cryogenic temperatures, the approach achieves up to a 26.89% average reduction in power consumption compared to a state-of-the-art cryogenic-unaware algorithm. This optimization enables large-scale standard-cell-based digital circuits to operate efficiently at cryogenic temperatures in crucial applications.

I. INTRODUCTION AND MOTIVATION

Cryogenic CMOS circuits describe circuits that operate at temperatures below 120 K. They are typically distinguished between operating at 77 K, which is the boiling point of liquid nitrogen (LN₂), and operating below 77 K with liquid helium (LHe) cooling down to 4.2 K. This distinction is made as LN₂ allows submerging a circuit in cheap liquid nitrogen for cooling, while LHe is much more expensive and frequently cooled in (vacuum) cryogenic cooling chambers to minimize the escape of costly He. Unfortunately, the majority of applications operate below 77 K such as quantum computers (4 K to 0.1 K) [1], magnetic resonance imaging (23 K to 4 K) [2], particle detectors like the *Large Hadron Collider* (4 K) [3], and space missions like the *James Webb Space Telescope* (27 K) [4].

All these circuits below 77 K operate under stringent power constraints, as cryogenic cooling at these deep cryogenic temperatures is very limited in its cooling power. For instance, when cooling below 1 K, LHe dilution refrigerators have cooling powers of 2 mW at 0.1 K, while zero-G cooling in space can solely cool 0.1 μW at < 0.1 K (92.6 mK) [5]. If the temperature is allowed to be higher, then cooling power ranges from 180 mW for optical laser (group-II–VI cadmium sulphide nanoribbons) cooling [6] or a power budget of 384 mW for regular CMOS control circuits of quantum computers at 3 K [7]. Note that these power numbers are merely stating how much thermal energy a cryogenic cooler can dissipate from the chamber, their overall electrical power

consumption is frequently > 1000 W [8]. Therefore, it is of utmost importance to reduce the power consumption of cryogenic CMOS circuits at the design stage already to operate within these constraints of cryogenic cooling.

Fortunately, semiconductor physics support this aim because, at cryogenic temperatures, leakage current of transistors decreases. Among other changes like a sub-threshold slope decrease and carrier mobility increase, this means that transistors operate more efficiently at cryogenic temperatures. For instance, Parihar *et al.* [9] showed a 99.999% reduction in leakage, while drive currents were roughly unaffected (+3%/−2.7% for NMOS/PMOS) at 10 K in 5 nm FinFET transistors.

To this end, to achieve cryogenic awareness at the design stage, the conventional logic design flow does not have to be completely re-invented from scratch. However, certain aspects close to the technology level require the introduction of novel objectives to enable a dedicated optimization for the cryogenic domain. Recent efforts in the field yielded a cryogenic transistor model and corresponding standard-cell library together with initial logic synthesis considerations [10]. Nevertheless, technology mapping, i. e., the process of generating technology-specific gate-level netlists from technology-independent logic descriptions, still offers room for improvement. While existing methods relied on minimizing (simulated) switching activity [11], a consideration of real static and dynamic power constraints at cryogenic temperatures has yet to be incorporated.

In this work, we propose a technology mapping approach, which takes a technology-independent logic network and a cryogenic standard-cell library as input and produces a technology-mapped gate-level netlist with significantly reduced power characteristics compared to the state-of-the-art (non-cryogenic) technology mapper [12]. Naturally, such an endeavor has to come at the cost of either circuit propagation delay or circuit area. Power can only be traded off against these other figure of merits of a circuit due to the exceptional capabilities of existing techniques. However, it is crucial, that these metrics are traded off at the Pareto front. The reduction in power consumption should cost the minimal increase of circuit delay, i. e., following the Pareto front, as shown in Fig. 1.

Experimental evaluations confirm that the technology mapping approach for cryogenic CMOS circuits, which we present in this work, reduces their power consumption by up to 26.89% on average compared to a state-of-the-art technique [12]. This enables these circuits to stay within the stringent power constraints of cryogenic cooling. The proposed solution opens the field of cryogenic operation to large-scale

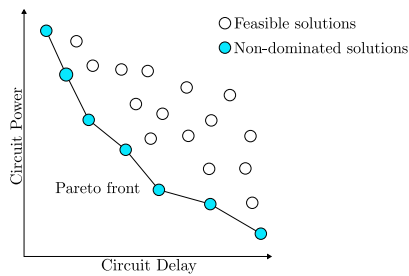


Fig. 1: Pareto front of area vs. delay in optimized circuits. Reducing one yields an increase in the other.

standard-cell-based digital circuits in areas such as quantum computing, magnetic resonance imaging, particle detectors, and space missions.

The remainder of this paper is structured as follows: Section II discusses the preliminary groundwork necessary for this work; Section III proposes the main contribution of this article, which is a cryogenic-aware technology mapper; Section IV presents results of an experimental evaluation against the state of the art; finally, Section V concludes the paper.

II. PRELIMINARIES

To keep this work self-contained, this section briefly discusses the state of the art in cryogenic CMOS circuitry and reviews the conventional logic design flow.

A. Cryogenic CMOS Circuitry

Cryogenic CMOS circuits are widely employed, such as in quantum computers [1], magnetic resonance imaging [2], particle detectors like the *Large Hadron Collider* [3], and space applications like the *James Webb Space Telescope* and other missions [13]. An overview of the many applications is given in [14]. Typical temperature ranges are 77 K (to be compatible with LN₂ cooling), 23 K (to exploit the superconductivity of NbTi [2]), and 4.2 K (to be compatible with LHe cooling).

To model the behavior of a circuit at these cryogenic temperatures, a cryogenic transistor model is required. FinFET transistors are typically modeled with the industry standard *Berkeley Short-channel IGFET Model – Common Multi-Gate* (BSIM-CMG, [15]). For cryogenic temperatures, the additional physical effects appearing at cryogenic temperatures have been incorporated by [16], which have presented a cryogenic BSIM-CMG model. Cryogenic model parameters have also been presented in the cryogenic transistor data reported in [9], which reports 5 nm FinFET transistor data from 300 K to 10 K.

Large-scale digital circuits are composed of *standard cells*, which are pre-implemented building blocks. Typical standard-cell libraries provide information on delay and power consumption for -40°C to 120°C (approximately 230 K to 400 K). However, recent works have proposed standard-cell libraries that have been characterized at cryogenic temperatures such as 10 K [10]. For each path through a standard cell (from input pin to output pin), the propagation delay and static/dynamic power was simulated for different signal slews (rate of change of voltage at the input pin) and output capacitance (load capacitance at the output pin). With

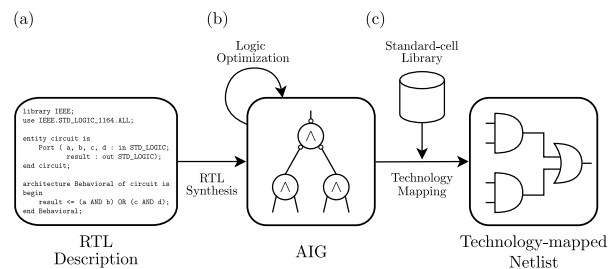


Fig. 2: Excerpt from the logic design flow consisting of RTL synthesis, logic optimization, and technology mapping.

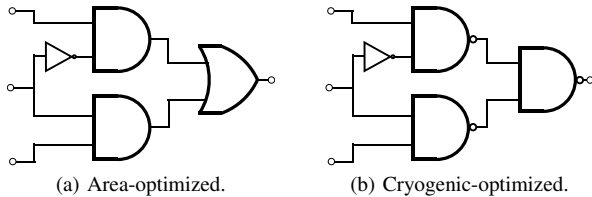
this information in standard-cell libraries, the construction of large-scale cryogenic digital circuits via logic synthesis and the analysis of circuits via *Static Timing Analysis* (STA) became possible for the first time [10].

B. Logic Design Flow

The logic design flow involves the conversion of a *Register-Transfer Level* (RTL) description to a gate-level netlist. This process, which is illustrated in Fig. 2, consists of three main steps: (a) *RTL synthesis*, (b) *logic optimization*, and (c) *technology mapping*. In RTL synthesis, a technology-independent logic network is created from a high-level architectural description. One commonly used technology-independent structure is the *And-Inverter Graph* (AIG, [17]). Logic optimization aims to improve the logic network with respect to technology-independent cost metrics. Consequently, AIGs are optimized using proxy criteria that are assumed to be beneficial for many technologies. Common criteria include size (number of nodes) and depth (number of levels), as they strongly correlate with chip area, critical path length, and power dissipation. Finally, technology mapping translates the logic network into a gate-level netlist exclusively using gates offered by the target technology, which are usually provided by a corresponding standard-cell library. In this work, we specifically focus on the technology mapping step, as the effects of cryogenic temperatures on delay and power are only reflected in this stage through CMOS technology information provided by a standard-cell library file, e. g., in *liberty* format.

III. TECHNOLOGY MAPPING FOR CRYOGENIC CMOS CIRCUITS

In order to properly realize cryogenic CMOS circuitry, a design flow similar to the one briefly reviewed above is required. While certain steps such as RTL synthesis may remain identical to conventional (i. e., non-cryogenic) circuitry, the technology-specific steps need to be re-visited. The endeavor requires cryogenic-aware synthesis using a dedicated library as well as corresponding technology mapping methods. While a first cryogenic standard-cell library as well as a corresponding synthesis approach addressing the former step have recently been proposed in [10], cryogenic-aware technology mapping remains an open problem thus far. In this work, we are addressing this issue. To this end, we do not propose to re-invent the wheel, but instead to re-use as much knowledge as possible from decades of development in the domain of technology mapping. In this section, we first review the state



	INV	AND	NAND	OR	(a)	(b)
Area	0.10	0.20	0.25	0.20	0.70	0.85
Power	0.10	0.35	0.30	0.35	1.15	1.00

(c) Simplified example cost excerpt.

Fig. 3: Implementations of a 2:1-MUX circuit optimized for different cost metrics.

of the art in technology mapping. Afterward, we describe the main changes we are proposing in order to make the flow suitable for cryogenic circuits.

A. Conventional Technology Mapping

For conventional technology mapping, to the best of the authors' knowledge, one of the most efficient algorithms is the one proposed in [12], which is implemented in *ABC* [18] as the command `map`. Using this algorithm as a representative, three internal steps are covered in the following: the creation of *Supergates*, *Simplified Boolean Matching*, and *Cover Selection*. In the following, these steps are briefly reviewed.

Supergates combine standard-cell library gates into a single-output network, treated as another standard cell, to reduce dependency on network structure and improve the overall solution [12]. They are formed iteratively in rounds, with the first round consisting exclusively of gates from the input library. Subsequent rounds create combinations of two supergates from the previous round, with a maximum bound on the input size. To handle the large number of supergates with the same Boolean function, dominance pruning is applied, i. e., equivalent Boolean functions are compared by their cost, and dominated supergates are excluded from the resulting library, which is organized in a hash table.

Example 1. Fig. 3 illustrates the results of a supergate creation. Both circuits depicted in Fig. 3a and 3b realize and, hence, represent a 2:1-MUX implementation. Simplified cost values for area and power are provided per gate in Fig. 3c. Considering area as the cost criterion (as is common in conventional technology mapping), the implementation in Fig. 3a possesses a total cost value of 0.70, which is less than the implementation in Fig. 3b with a total cost value of 0.85. Accordingly, the latter would be excluded from the supergate library, because it is dominated in terms of area costs. Therefore, only cost-optimal supergates are available for subsequent steps in the technology mapper.

Boolean Matching matches previously constructed supergates onto cuts, allowing the replacement of a part of the input logic network with a supergate, i. e., a selection of standard-cell library gates. A cut of node n (called *root*) represents a group of nodes referred to as leaves, such that

every path from a primary input to node n must traverse through at least one of these leaves. The cut is k -feasible if its *cardinality* (number of leaves) does not surpass the value k .

Before performing Boolean Matching, all k -feasible cuts for each node are calculated, and their respective Boolean functions are determined. In the context of [12], only 5-feasible cuts are considered, limiting supergates to 5 inputs. This restriction is necessary because the number of cuts increases significantly with higher values of k . Furthermore, the Boolean function of the cut and its NPN-equivalence class are computed and matched with a supergate, expanding the search space. Two functions are NPN-equivalent if they can be transformed into each other by permuting the input order and optionally negating primary inputs and outputs.

For an efficient implementation, *Simplified Boolean Matching* can be used which avoids NPN-equivalence checking in the body of the mapper and, hence, speeds up the process [12]. Therefore, all 5-input Boolean functions are precomputed by permuting all library supergates and adding them to the supergate hash table. When finding a supergate from the library to match the Boolean function of a cut, the mapper refers to the supergate hash table and selects the most suitable match based on a cost function.

Example 2. Consider again the circuits shown in Fig. 3. Each gate in both implementations represents one cut in the original network. Simplified Boolean matching looks up supergates and their respective NPN-equivalent supergates for each cut, and then maps the gate with the better cost. For instance, the Boolean functions AND and NAND are NPN-equivalent. When considering their respective costs from Fig. 3c, the AND gate is preferred due to its smaller area (0.20 compared to 0.25). This process is repeated for each cut, resulting in an area-optimal cover for the circuit in Fig. 3a.

Cover Selection creates a comprehensive *cover* by selecting the best-matched cuts for the entire network, based on a specific cost criterion. In subsequent iterations, as the circuit is optimized according to other cost metrics, the mapper can choose if the newly determined solution is better or worse than the present mapping.

This approach is sensible because some cost functions rely on heuristics, and exploring different heuristics may lead to discovering better solutions randomly. It also enables finding trade-off solutions. For conventional technology mapping, in the first round, the best delay mapping is identified (an optimal solution can be found with dynamic programming), and in subsequent steps, when optimizing for area, the cover selection can be restricted to optimize cuts only if the critical path is not extended.

Additionally, both individual and interconnected influences of cuts are considered from a global perspective when selecting cut implementations based on their cost. For area, this can be done by utilizing a heuristic called *areaflow* [19], which evaluates the area of individual cuts and allows a cut's area to influence the cost of connected cuts at its output, leading to a more comprehensive mapping consideration.

Example 3. Consider the AIG depicted in Fig. 4. The most area-efficient mapping of this sub-network may be found by taking the cut_1 rooted at node n_3 . However, this would imply

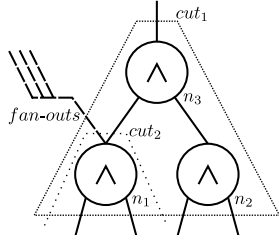


Fig. 4: Cut decisions based on different heuristics.

that node n_1 must be duplicated for its other fan-outs. Instead, the cuts (marked as cut_2) can also be chosen to be rooted at node n_1 . This approach ensures that the node will not be duplicated, preventing the inclusion of additional gates in the final network. This is beneficial as additional gates are likely to increase the overall area of the circuit. This solution can be found using a global heuristic like *areaflow*.

Overall, conventional technology mapping is key in order to determine a circuit representation, which is optimized for a particular technology and its cost objectives. However, corresponding methods and tools (such as *ABC* [12] as a common and well established example) are designed to conduct all the steps reviewed above for a pre-defined static cost metric—mainly area and delay thus far. When considering cryogenic applications, area is insignificant while delay cannot be entirely disregarded. Thus, methods for conventional technology mapping lead to sub-par results when applied for the design of cryogenic circuits. Motivated by this, the remainder of this section introduces new cost metrics as well as a meaningful application for all three steps reviewed above—resulting in a cryogenic-aware technology mapper.

B. Cryogenic-specific Cost Metrics

To ensure cryogenic awareness, it is crucial to focus on the dynamic power consumption-related data. As mentioned in Section I, leakage can be neglected in the cryogenic domain. Therefore, an average dynamic power consumption denoted as $p_{dyn,avg}$ is proposed, taking into account the costs associated with gates. Furthermore, the introduction of *dynflow*, which incorporates the first cost function, allows a mapper to obtain a more global perspective of the cryogenic cost by also considering the network structure.

For the average dynamic power consumption, the dynamic power losses of a gate, which consist of the cell internal power and net switching power, are taken into account. These two metrics illustrate the power losses that occur in a cell when there is a change at an input pin. Specifically, the net switching power refers to the losses when the switching input pin triggers a change at the output pin. Otherwise, the power losses are attributed to the cell internal power.

In order to provide cryogenic awareness, the average dynamic power consumption is computed for each cell in the provided standard-cell library. Taking into account the dependence of both, net switching power and cell internal power, on the input signal, the standard-cell library presents multiple values corresponding to different input signal slews. Additionally, the power losses for net switching power are dependent

on various output cell capacitances. For simplicity, $p_{dyn,avg}$ represents an average power consumption, encompassing the respective variables for net switching power and cell internal power. The average cell internal power and net switching power for an input pin i are defined as follows:

$$p_{avg,int}(i) = \left(\sum_{k=0}^{n_{slew}} (p_{int,k}(i)) \right) \cdot \frac{1}{n_{slew}} \quad (1)$$

$$p_{avg,net}(i) = \left(\sum_{l=0}^{n_{slew}} \sum_{m=0}^{n_{cap}} (p_{net,l,m}(i)) \right) \cdot \frac{1}{n_{slew} \cdot n_{cap}} \quad (2)$$

Assuming equal switching activity on each pin, gates with more pins tend to consume more power than gates with fewer pins.

Example 4. Considering the power consumption of an AND4 gate with four inputs, compared to the power consumption of an AND2 gate with two inputs, assuming an equal switching probability at each pin, the AND4 gate is twice as likely to see a switching activity at one of its input pins. This should not be confused with the output pin's switching probability. For an AND4 gate, output pin switching is less frequent, making internal cell power a more significant factor in dynamic power consumption than net switching power. These aspects are already reflected in the values p_{int} and p_{net} .

Therefore, the number of pins is directly linked to the power consumption of the gate. Based on this information, we present the initial cryogenic-aware cost function, which is the average dynamic power consumption $p_{dyn,avg}$. It is defined as the sum of the power consumption of the input pins of a gate g :

$$p_{avg,dyn}(g) = \sum_{r=0}^{n_{pins}} (p_{avg,int}(r) + p_{avg,net}(r)) \quad (3)$$

In addition to the average power, which can be used to compare individual cuts, we propose a global heuristic derived from *areaflow* as introduced in [12]. More precisely, we define the *dynamic power flow* as follows:

$$dynflow(c) = p_{dyn}(c) + \sum_{s=0}^{n_{leaves}} \frac{dynflow(leaf_s(c))}{n_{fanout}(leaf_s(c))}, \quad (4)$$

with $p_{dyn}(c)$ being the dynamic power of an individual cut c and $leaf_i(c)$ being the i -th leaf of c . Each leaf is contributing its own *dynflow* divided by the number of its fan-outs to c .

These cost metrics properly represent the optimization objectives for a technology mapping method aiming at cryogenic circuits. Based on that, the corresponding steps reviewed above can be adjusted for a cryogenic-specific consideration as described in the following.

C. Supergates

As already mentioned in Section III-A, supergates are conventionally grouped by their Boolean function in a hash table. When they are found to be dominated by another supergate, they are not included in the supergate library. In the cryogenic domain, instead of area and delay, our objective is to select the supergate with the lowest dynamic power consumption.

To achieve this, we need to determine the dynamic power consumption of each supergate for comparison. The dynamic power consumption of a supergate is defined as the sum of the average dynamic power consumption of the individual gates it comprises. Due to this step, we provide (Simplified) Boolean Matching with a cryogenic-aware supergate library.

Example 5. Consider again Fig. 3 and the previous Example 1. When comparing which gate should be included into the supergate library, this time a cryogenic-aware decision can be made. When adding up the $p_{avg,dyn}$ -values for the sub-network, the circuit in Fig. 3a results in power costs of $0.1 + 2 \cdot 0.35 + 0.35 = 1.15$ and the one in Fig. 3b results in costs of $0.1 + 3 \cdot 0.30 = 1.00$. Hence, the latter implementation is clearly favorable. This outcome is contrary to the area cost, which would favor the former implementation (0.70 compared to 0.85).

D. Simplified Boolean Matching

During (Simplified) Boolean Matching, the Boolean functions of cuts are matched against supergates, aiming to discover an efficient implementation of the network concerning a specific cost metric. As mentioned above, Boolean Matching also allows the selection of NPN-equivalent classes. When matching the cuts based on cryogenic costs, it provides the cover selection with the relevant information to also make cryogenic-aware decisions.

Example 6. Consider again Example 2 and Fig. 3. Using the new, cryogenic-specific, cost functions (rather than the area-focused ones from conventional technology mappers) leads to a different assessment of the dominance of the two supergates. Comparing the power costs provided in Fig. 3c for the AND gate (0.35) to those of the NAND gate (0.30), the NAND gate is now selected. Hence, the implementation in Fig. 3b is preferred.

E. Cover Selection

When choosing a cover, the comparison of cut costs relies on evaluating them through a cryogenic-aware cost function. This essential procedure can be supported by techniques such as *dynflow*, which considers not just the $p_{dyn,avg}$ but also the number of fan-outs a gate has, and splits its cost upon them. It has to be noted, that the fan-out of a node only persist after mapping if this node is used as a root node. Otherwise, if the multi-fan-out node is covered in a cut, it has to be duplicated. This means that this heuristic tries to use nodes with fan-outs as root nodes to avoid node duplication to a certain degree. For a delay optimal mapping on the other hand, node duplication can be highly beneficial in certain cases.

Example 7. For the AIG shown in Fig. 4, the heuristic *dynflow* enables a global comparison of cuts, considering both the network structure and the cost $p_{dyn,avg}$. Considering again the cuts rooted at node n_3 and node n_1 , with *dynflow*, a cryogenic-aware decision can be made, determining whether node duplication is advantageous or should be avoided.

Given that *dynflow* is employed to choose more optimal cuts, subsequent to the cut selection, the individual cuts are once again matched with respect to $p_{dyn,avg}$.

TABLE I: Experimental comparison against *ABC*'s map command [12] considering the power difference at 1 GHz

BENCHMARK [20]	IMPROVEMENT OVER S-O-T-A [12] IN %							
	Delay-Power-Aware				Power-Inf.-Delay			
	<i>NSP</i>	<i>CIP</i>	P_{tot}	<i>PDP</i>	<i>NSP</i>	<i>CIP</i>	P_{tot}	<i>PDP</i>
adder	-12.43	58.43	17.85	27.57	-3.18	67.37	26.97	-28.33
arbiter	1.08	-0.14	0.79	-18.03	8.63	2.31	7.19	-14.98
bar	8.80	17.87	11.88	18.16	18.16	38.49	25.07	31.59
cavlc	3.20	5.75	4.03	-23.54	0.31	12.91	4.41	-22.45
ctrl	-4.93	6.33	-2.13	-47.87	-10.07	23.20	-1.83	-47.18
dec	0.61	5.92	1.25	0.04	-7.02	22.41	-3.36	3.17
hyp	-4.42	39.50	16.20	13.29	18.14	58.25	36.97	19.72
i2c	1.45	15.71	5.14	-16.53	0.85	17.12	5.06	-9.15
int2float	-4.07	5.44	-1.10	-39.35	-3.43	10.38	0.89	-58.53
log2	5.47	29.20	14.64	-8.59	44.65	64.15	52.17	32.14
max	6.60	15.08	9.30	2.38	19.86	27.67	22.39	-3.13
mem_ctrl	0.30	14.93	5.29	1.27	4.14	19.47	9.40	8.11
multiplier	-1.56	32.25	12.65	6.88	33.10	62.39	45.38	16.71
priority	15.33	21.69	17.14	13.03	54.51	47.26	52.42	31.26
router	0.59	56.04	26.64	40.98	13.84	66.30	38.50	2.68
sin	22.55	41.44	29.40	21.46	32.92	49.96	39.12	25.38
sqrt	-4.55	38.66	12.43	45.17	56.59	71.48	62.33	83.76
square	-1.49	54.60	24.97	17.07	5.97	58.74	30.82	-49.21
voter	-2.46	9.50	1.64	-7.26	53.15	64.35	57.00	51.21
<i>Average</i>	1.58	24.64	10.95	2.43	17.95	41.27	26.89	3.83

NSP and *CIP* are the net switching power and cell internal power respectively. P_{tot} is the total power and *PDP* is the power-delay-product.

Additionally, cover selection can choose between the present cover and the newly obtained cover. Starting from a delay-optimal cover and considering all the matches found for power optimization in the second run, the mapper is able to take both possibilities into account. This enables the mapper to choose—informed by a cost function—which cut to prefer. If delay optimality needs to be preserved, the mapper can decide to only select the power-optimal solutions that do not negatively impact the critical path. Otherwise, the mapper can disregard delay completely. This realization yields two modes of operation for the proposed cryogenic-aware technology mapper: In the (1) *Delay-Power-Aware* mode, the mapper optimizes for dynamic power consumption without impacting the critical path of the network. On the other hand, the (2) *Power-Inf.-Delay* mode freely trades delay for dynamic power savings without limitations.

Overall, cover selection is responsible for choosing the final mapping and has the capability to take into account multiple variables. It does not only help to find a more globally-optimal solution for the cryogenic domain, but it also offers the designer various choices for optimization.

IV. EXPERIMENTAL EVALUATION

The technology mapper proposed in the previous sections takes the considerations of the cryogenic domain into the logic synthesis process. To validate the effectiveness of this technology mapper, evaluations against the conventional technology mapping (reviewed in Section III-A) have been conducted. In this section, the correspondingly obtained results are summarized and discussed. To this end, we first provide an overview of the used setup. Afterwards, we present and discuss the obtained results.

A. Experimental Setup

In our evaluations, the cryogenic-aware standard-cell library from [10] is utilized. As reference, the *EPFL Benchmark Suite* [20] has been considered, which offers a diverse set of

combinational logic networks widely used in the logic synthesis community. The proposed cryogenic-aware technology mapper was implemented in C on top of *ABC*'s commands `map` and `read_lib` based on the concepts provided in Section III.

The output of the proposed flow are mapped netlists in Verilog format, which have been generated via the new `map` command and written via *ABC*'s `write_verilog` command. The generated netlists have been entered into *Synopsys PrimeTime*, a commercial tool used for STA, which provides information about the resulting power consumption and delay values for each of the circuits. To ensure a fair comparison, the power consumption is evaluated independently of the delay at a fixed clock frequency of 1 GHz.

B. Obtained Results and Discussion

The results are generated using the EPFL benchmarks and the cryogenic standard-cell library, enabling a comparison between the conventional and cryogenic-aware technology mappers. Moreover, the latter includes the two proposed modes. Table I summarizes the obtained results; for each benchmark, the improvements in net switching power, cell internal power, as well as the total power improvement at 1 GHz, are listed. On average, net switching power is about three times higher than cell internal power, while the latter is much more affected by the new mapper. As discussed, leakage can be neglected at cryogenic temperatures. Additionally, the power-delay-product (PDP) is presented. The last row provides the averages for all metrics.

For the *Delay-Power-Aware* mode, the average power losses at 1 GHz could be reduced by 10.95 % on average. For 2 out of 19 cases, the power losses are increasing (worst case *ctrl*). Regarding the delay, the PDP is decreasing slightly by 2.43 %. The more unrestricted *Power-Inf.-Delay* mode even results in average power savings of 26.89 %. The increase in power losses can also be observed in 2 cases, with the case of *dec* being the worst. Additionally, the PDP decreases by 3.83 % on average.

Two significant conclusions can be drawn from these results: first of all, considering only power savings, the proposed technology mapper achieves an average power reduction of 26.89 % with up to 62.33 % (in case of *sqrt*) in the *Power-Inf.-Delay* mode. These are immense improvements for the cryogenic domain. Secondly, upon examining the delay aspect, both proposed modes of the technology mapper demonstrate the ability to trade delay for a reduction of power loss, nearly in a one-to-one relationship. This can be observed by the delay-power product, which even improves by 2.43 % and 3.83 % for the respective modes. Thus, the results confirm the effectiveness of the proposed technology mapper in generating solutions proximate to the Pareto-optimal front, as discussed in Section I. This approach provides the designer with two modes to assess the trade-off between delay and power loss. Ultimately, these achieved results serve as validation for the cryogenic awareness of the proposed technology mapper.

V. CONCLUSION

This work introduces the first cryogenic technology mapper, considering essential steps in technology mapping, namely

Supergate Creation, Boolean Matching, and Cover Selection, to integrate cryogenic considerations. Prioritizing dynamic power dissipation as the primary optimization target while also considering delay, we devised a mapping approach tailored for cryogenic environments. The experimental evaluation provides compelling confirmation of its effectiveness. The findings from this research serve as a motivation towards the development of a comprehensive cryogenic-aware logic synthesis procedure. By developing a logic synthesis flow for cryogenic technologies, we unlock the potential of highly specialized technologies operating at cryogenic temperatures, promising the compliance with strict power constraints.

REFERENCES

- [1] B. Patra, R. M. Incandela *et al.*, "Cryo-CMOS circuits and systems for quantum computing applications," *IEEE Journal of Solid-State Circuits*, vol. 53, no. 1, pp. 309–321, 2017.
- [2] D. H. Johansen, J. D. Sanchez-Heredia *et al.*, "Cryogenic preamplifiers for magnetic resonance imaging," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 12, no. 1, pp. 202–210, 2017.
- [3] D. Braga, S. Li, and F. Fahim, "Cryogenic Electronics Development for High-Energy Physics: An Overview of Design Considerations, Benefits, and Unique Challenges," *IEEE Solid-State Circuits Magazine*, vol. 13, no. 2, pp. 36–45, 2021.
- [4] G. Bagnasco, M. Kolm *et al.*, "Overview of the near-infrared spectrograph (NIRSpec) instrument on-board the James Webb Space Telescope (JWST)," in *Cryogenic Optical Systems and Instruments XII*, vol. 6692. SPIE, 2007, pp. 174–187.
- [5] H. Zu, W. Dai, and A. De Waele, "Development of dilution refrigerators—a review," *Cryogenics*, vol. 121, p. 103390, 2022.
- [6] J. Zhang, D. Li *et al.*, "Laser cooling of a semiconductor by 40 kelvin," *Nature*, vol. 493, no. 7433, pp. 504–508, 2013.
- [7] X. Xue, B. Patra *et al.*, "CMOS-based cryogenic control of silicon quantum circuits," *Nature*, vol. 593, no. 7858, pp. 205–210, 2021.
- [8] H. Cao and H. ter Brake, "Progress in and Outlook for Cryogenic Microcooling," *Phys. Rev. Appl.*, vol. 14, p. 044044, Oct 2020. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevApplied.14.044044>
- [9] S. S. Parihar, V. M. van Santen *et al.*, "Cryogenic CMOS for Quantum Processing: 5-nm FinFET-Based SRAM Arrays at 10 K," *IEEE Transactions on Circuits and Systems*, 2023.
- [10] V. M. van Santen, M. Walter *et al.*, "Design Automation for Cryogenic CMOS Circuits," in *Proceedings of DAC'23*, 2023.
- [11] S. Jang, K. Chung *et al.*, "A power optimization toolbox for logic synthesis and mapping," 2009.
- [12] S. Chatterjee, A. Mishchenko *et al.*, "Reducing structural bias in technology mapping," *TCAD*, 2006.
- [13] R. Patterson, A. Hammoud *et al.*, "Electronic components and systems for cryogenic space applications," in *AIP Conference Proceedings*, vol. 613, no. 1. American Institute of Physics, 2002, pp. 1585–1590.
- [14] R. K. Kirschman, "Low-temperature electronics," *IEEE Circuits and Devices Magazine*, vol. 6, no. 2, pp. 12–24, 1990.
- [15] M. V. Dunga, C.-H. Lin *et al.*, "BSIM-CMG: A compact model for multi-gate transistors," in *FinFETs and Other Multi-Gate Transistors*. Springer, 2008.
- [16] G. Pahwa, P. Kushwaha *et al.*, "Compact Modeling of Temperature Effects in FDSOI and FinFET Devices Down to Cryogenic Temperatures," *IEEE Transactions on Electron Devices*, vol. 68, no. 9, pp. 4223–4230, 2021.
- [17] A. Kuehlmann and F. Krohm, "Equivalence checking using cuts and heaps," in *DAC*, 1997.
- [18] R. Brayton and A. Mishchenko, "ABC: An Academic Industrial-Strength Verification Tool," in *CAV*. Springer, 2010, pp. 24–40.
- [19] V. Manohararajah, S. D. Brown, and Z. G. Vranesic, "Heuristics for area minimization in LUT-based FPGA technology mapping," *IEEE Transactions on CAD*, 2006.
- [20] L. Amarú, P.-E. Gaillardon, and G. De Micheli, "The EPFL combinational benchmark suite," in *IWLS*, 2015.