Selecting the Optimal Energy Point in Near-Threshold Computing

Sami Salamin, Hussam Amrouch, and Jörg Henkel
Karlsruhe Institute of Technology, Chair for Embedded Systems (CES), Karlsruhe, Germany
{sami.salamin; amrouch; henkel}@kit.edu

***Abstract*—Near-Threshold Computing (NTC) has recently emerged as an attractive paradigm as it allows devices to operate close to their optimal energy point (OEP). This work demonstrates, for the first time, that determining where the OEP of a processor exists is challenging because standard cells, forming the processor's netlist, unevenly profit w.r.t power and also unevenly degrade w.r.t delay when the voltage approaches the near-threshold region. To precisely explore, at design time, where OEP is, we create voltage-aware cell libraries that enable designers to seamlessly employ the standard tool flows, even they were not designed for that purpose, to perform voltage-aware timing and power analysis. Besides determining where the OEP is, we also demonstrate how providing logic synthesis tool flows with voltage-aware cell libraries results in a 35% higher performance at NTC. In addition, we investigate how the performance loss at NTC can be compensated through parallelized computing demonstrating, for the first time, that the OEP moves far from NTC as the number of cores increases. Our proposed methodology enables designers to select the maximum number of cores along with the optimal operating voltage jointly in which a specific power budget is fulfilled. Finally, we show how voltage-aware design for parallelized NTC provides [40%-50%] performance increase compared to traditional (i.e., voltage-unaware design) parallelized NTC.**

I. Introduction

The increasing trend to maximize the energy efficiency of devices has pushed NTC to the forefront of promising solutions because it enables circuits to operate close to their optimal energy point (OEP) – this holds even more for Internet of Thing (IoT) devices, where the lifetime of the battery is a key limiting factor. The complexity of IoT devices is steadily increasing as a variety of services (e.g., sensing/processing data, exchanging information with cloud, etc.) is increasingly required. Therefore, there is an inevitable need to recover the large performance loss that incurs at NTC. As a matter of fact, reducing the operating voltage (V_{dd}) of a circuit leads to a significant reduction in its total power due to the quadratic saving in dynamic power ($P_{dynamic}$) along with the exponential saving leakage power ($P_{leakage}$) (see Eq. 1 [1]). The drawback of down-scaling V_{dd} is, however, the quadratic increase in transistor delay (see Eq. 2). As a result, the maximum delay of the critical path of the circuit (t_{CP}) becomes larger, leading to a lower operating frequency ($freq$). Hence, a considerable performance loss incurs as V_{dd} approaches the NTC region.

$$P_{dynamic} \approx C_{eff} V_{dd}^2 f \ , \ P_{leakage} \approx V_{dd} K_1 e^{K_2 V_{dd}} \tag{1}$$

$$freq = \frac{1}{t_{CP}} \ ; \ t_{CP} = \sum_{T_i \in CP} t_{T_i} \ ; \ t_{T_i} \propto \frac{1}{(V_{dd} - V_{th})} \tag{2}$$

Where C_{eff} is average switched capacitance, K_1 and K_2 are fitting parameters [1]. t_{T_i} is delay of transistors that contribute to t_{CP} and V_{th} is the threshold voltage of transistor.

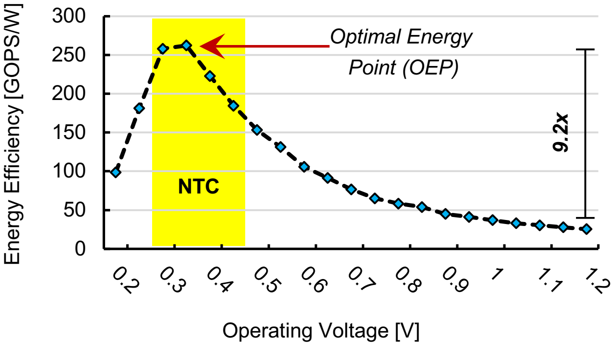

Fig. 1. Impact of voltage scaling on energy showing how the maximum energy efficiency exists within the near-threshold region. The Discrete Cosine Transform (DCT) circuit, often used in image encoding, is considered in this analysis at the 45nm technology node.

Optimal Energy Point (OPE): Given such a wide voltage scaling potential, it is essential to determine the V_{dd} at which the energy efficiency is *maximum* because the energy of any circuit is the product of its both power and delay. The total energy per cycle reduces with V_{dd} until an inflection point at which it starts to exponentially increase as the leakage energy per cycle compensates the reduction in dynamic energy per cycle [2]. The V_{dd} where such an inflection occurs typically represents the OEP at which the energy efficiency is maximum (see Fig. 1). As shown, operating the circuit at 0.35V results in a maximum energy efficiency (which is defined as the maximum frequency divided by the worst-case power [2]) of 9.2x compared to 1.2V in the super-threshold region.

Parallelized Near-Threshold Computing: The key drawback of NTC is the significant performance loss. For instance, reducing V_{dd} from 1.2V to 0.4V leads to 141x larger delay (i.e. lower frequency). Such a significant reduction in $freq$ considerably limits the possible services that a processor may offer at NTC. To compensate, additional cores can be included in which multiple tasks are parallelized. However, finding the OEP then becomes more complicated. The main reason is that the Amdahl serial factor [3] (i.e. the percentage of the sequential part) of the executed applications akin to different limitations within the parallel execution (details in Section V) plays a major role in defining the gain from parallelism and thus the required number of cores which is needed to fulfill the required performance. In addition, despite the impact of using more cores on compensating the performance loss, a higher number of cores directly results in higher power consumption. *In this work, we investigate the impact of parallelized NTC on shifting OEP out of the near-threshold region.*

978-3-9819263-2-3/DATE19/©2019 EDAA

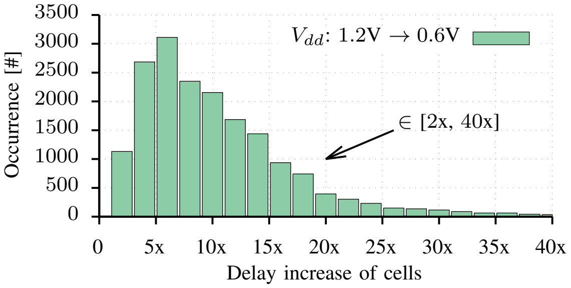

Fig. 2. The same voltage reduction *unevenly* increases the delay of standard cells. A 45nm standard cell library has been here analyzed.

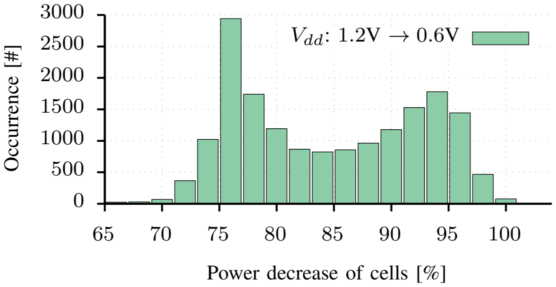

Fig. 3. The same voltage reduction *unevenly* reduces the power of standard cells. A 45nm standard cell library has been here analyzed.

II. Key Challenges behind Determining the Optimal Energy Point (V_{opt})

To understand the impact of voltage down-scaling on the energy of a processor, we investigate how V_{dd} reduction influences the delay and power of standard cells at the 45nm technology node. We present in Fig. 2 how a voltage reduction from 1.2V to 0.6V increases the delay of cells. As shown, the same V_{dd} reduction *unevenly* increases the delay of cells. While the delay of some cells is increased by only around 2x, the delay of other cells is increased by 10x and even up to 40x. Therefore, when studying the impact of voltage reduction on complex circuits like a full processor, it is quite difficult to accurately estimate the overall impact of voltage reduction on the processor's delay. This is because standard cells, which form the processor's netlist, will be unevenly affected by the same V_{dd} reduction. Hence, every cell within the critical paths of the circuit will contribute its *own* delay increase to the over all delay increase. Analogous to the previous analysis, we present in Fig. 3 the impact that a voltage reduction from 1.2V to 0.6V has on the power of cells. As shown, voltage reduction also *unevenly* impacts on the power of cells. While the power of some gates will be reduced by around 70%, the power of other cells may be reduced by more than 90%. All in all, the large variance in the power reduction of cells and in the delay increase of cells under the same voltage reduction makes estimating accurately how the energy of circuit will exactly profit from voltage scaling ambiguous and as a result, selecting where the OEP exists is challenging. This even holds more when a complex circuit consisting of a numerous number of cells is analyzed where every cell within the netlist will be differently affected by the same V_{dd} reduction.

Our novel contributions within this paper are as follows: We create voltage-aware cell libraries at the 45nm technology node [4]. They contain the detailed delay and power information of every cell covering the entire voltage range: starting from the super-threshold (i.e. 1.2V) all the way down to the sub-threshold (i.e. 0.2V) region. We deploy them within standard tool flows (Synopsys) to perform voltage-aware timing and power analysis in order to find the OEP of circuit. We demonstrate how using voltage-aware cell libraries during the logic synthesis tool provides circuits that exhibit higher performance in both single-core and parallelized NTC. We also demonstrate how voltage-aware cell libraries are prerequisite to selecting the OEP in the scope of parallelized NTC, revealing that the OEP is not necessarily within the near-threshold region and it may move out of it.

III. Related Work

NTC has been proposed in industrial approaches as well as in research. Intel presented in [5] its processor based on the IA-32 architecture with the capability to operate at NTC. Intel implemented multiple independent voltage domains to separate logic from SRAM-based components. This was necessary to ensure that the V_{min} of the memory does not limit the V_{min} of the logic core. Parallel Ultra Low Power Processor (PULP) [6] has been also recently introduced based on the open-source RISC-V processor for NTC operation. PULP shows a peak efficiency of 67MOPS/mW when it is fabricated at the 65nm technology node and up to 193MOPS/mW when it is fabricated using the low-power 28nm FD-SOI technology node. Optimizing netlists w.r.t degradation effects was proposed in [7], [8], for aging and temperature effects, respectively, but not for voltage. In [9], authors studied 7nm FinFET technology with respect to voltage scaling and compared it to planar MOSFET technologies. The results showed that FinFET provides 8.6x higher energy efficiency at NTC compared to the nominal operating voltage leading to 4.8x gain compared to 20nm planar MOSFET. The analysis has been done using a chain of thirty-one Inverters to investigate the impact of voltage reduction on delay and power towards analyzing the energy efficiency at NTC. A similar work based also on a chain of thirty-one Inverters has been presented in [10] to study parallelized NTC. [11] also compared FinFET technology with planar MOSFET with respect to energy for only a couple of different voltages (e.g., 0.3V, 0.45V, 0.8V and 1.1V) and using simple circuits such as Inverter chain, 16-bit adder and 16-bit multiplier. The study, and similar to [9], also showed that FinFET provides a better energy efficiency at NTC compared to planar CMOS technologies. [12] studied NTC with respect to approximate computing in which applications can tolerate errors. The work focuses on re-sizing channel length of nMOS transistors to avoid large transistors within the pull-up network. A minimum set of cells (i.e. INV, NAND, NOR, DFFR and DFFS) under varied voltages was analyzed. It is noteworthy that the impact of voltage reduction on standard cells is non-uniform and the variance is considerable as demonstrated in Figs. (2 and 3). Hence, studying only a few of cells (e.g, [11], [12]) and/or analyzing a chain of inverters (e.g. [9], [10] is insufficient to investigate where the OPE exists – especially in complex designs like a full processor.

Distinction from existing state of the art: we propose a methodology, which is compatible with existing EDA tool flows, that allows the investigating the OEP in both *single* and *parallelized* NTC in the scope of a sophisticated processor such as the Rocket CPU from Berkeley [13], which is an industry-competitive processor. This is unlike state of the art that considers relatively simple circuits like adders, multiples [11] or just a chain of inverters [9], [10].

IV. Voltage-Aware NTC Design

Standard cell libraries are typically developed and optimized to operate in the super-threshold region and they are far away from the near-threshold and sub-threshold regions. A standard cell library covers only a couple of discrete voltages that correspond to corner cases like fast-fast, slow-slow, and typical-typical corners. Relying on the Multi-Corner Multi-Mode (MCMM) analysis in which the delay and power of circuits are estimated by the EDA tool through the interpolation between the available corners suffers from a high inaccuracy. Note that, for a V_{dd} out of the voltage range exists between the available corners, the tool is unable to do any delay/power estimations because extrapolation is not possible. Therefore, to achieve our goal of accurately selecting the OEP, it is inevitable to create voltage-aware cell libraries in which the entire operating voltage range is fully covered (i.e. from the nominal voltage all the way the sub-threshold voltage). In the following, we demonstrate first how our proposed voltage-aware cell libraries are created along with how they can be employed to perform delay and power analysis. Afterwards, we explain our proposed idea of voltage-aware logic synthesis in which we consider the impact of voltage reduction on cells' delay during synthesis in order to obtain circuits that exhibit smaller delays (i.e. better performance).

Voltage-Aware Cell Libraries: In this work, we target the 45nm technology node. However, our proposed process is not limited to a specific technology and it can be analogously applied to any other nodes. We employ the SPICE netlist of different combinational and sequential cells from the 45nm open-cell library from Nangate [14]. To model the electrical characteristics of pMOS and nMOS transistors, we employ the Predictive Technology Modeling (PTM) [15] for the high-performance 45nm technology. To model the dependencies of MOSFET's parameters on voltage, we employ the BSIM model [16], which is the industry standard compact MOSFET model. Finally, we employ the HSPICE circuit simulator to accurately measure the delay and power of every standard cell under the impact of voltage reduction. To take the impact of the operating conditions into account, we consider 7 input signal slews along with 7 output load capacitances, which is typical in both industrial and academical cell libraries. The required range of input signal slew which needs to be analyzed at a specific voltage is determined based on the output slope of the fastest cell with minimum number of fanout (i.e. one), and the slowest cell with maximum fanout (i.e. 20). For instance, at $V_{dd} = 0.5$ the input signal slew is from 30ps to 10.92ns. To determine the required range of output load capacitance, which needs to be considered, the minimum number of fanout cell having least input capacitance and the maximum number of fanout cells having largest input capacitance is used. For instance, at $V_{dd} = 0.5$V the output load capacitance is from 0.5fF to 20fF. All the measured 49 (i.e. 7×7) delays of every cell are then stored within a lookup table per both timing and power information based on the standard "*liberty*" format. Hence, our created cell libraries are compatible with commercial tool flows like those that are provided by major known vendors and that are used commercially throughout. The employed library creation flow is similar to the proposed flow in [7] and [8], which targeted aging and temperature degradations, respectively. Based on the aforementioned process of library creation, we create cell libraries for the voltage range from 1.2V to 0.2V (with a step of 0.05V), i.e. 21 voltage-aware cell libraries in total are created. This enables us to precisely explore the impact of V_{dd} reduction on the energy of circuits from the super-threshold all the way down to the sub-threshold region.

Voltage-Aware Timing and Power Analysis: By creating such voltage-aware cell libraries, which are compatible with existing commercial tool flows , we propose to leverage the mature optimization algorithms of such tools (e.g. Synopsys and Cadence), evolved over more than a decade, to *accurately* and *automatically* estimate the energy consumption of any circuit (regardless of its complexity) under the entire range of V_{dd}. To achieve that, we first synthesize the RTL design at the nominal $V_{dd} = 1.2$V. Then, we iteratively perform voltage-aware delay and power analysis for the obtained netlist at every voltage step using our created voltage-aware cell libraries. This allows us to accurately estimate the maximum delay of circuit and its total power consumption (i.e. both leakage and dynamic power). This, in turn, enables us to calculate the consumed energy at every voltage step and hence determine how the energy efficiency improves while voltage reduces towards precisely selecting where the OEP occurs. To perform voltage-aware delay and power analysis at a specific voltage, we first merge all the voltage-aware cell libraries which have been created into one cell library along with adding the proper annotation of cells, which indicates to the corresponding voltage. After synthesizing the RTL design and thus obtain the resulting gate-level netlist, we annotate every cell within the netlist to have an index that corresponds to the wanted V_{dd} at which the analysis needs to be performed. Then, we provide our merged voltage-aware cell library together with the annotated netlist to the standard tool flows to perform timing and power analysis. Note that because the impact of a voltage reduction on delay of cells is non-uniform (see Fig. 2), the path that was originally critical at the nominal voltage will most likely not remain critical when V_{dd} is reduced and another path that was originally non-critical may takes its role at the lower voltage. Therefore, analyzing the entire circuit's netlist, as our methodology proposes, is necessary.

Voltage-Aware Logic Synthesis: As mentioned in Section I, operating circuits at NTC leads to a significant loss in performance due to the large delay increase when V_{dd} approaches NTC. To mitigate that, we propose in this work to bring voltage awareness to the logic synthesis tool and thus perform voltage-aware logic synthesis in which the synthesis tool considers the impact of voltage reduction early during the logic synthesis. As demonstrated in Fig. 2, the same voltage reduction has a non-uniform impact on the delay of cells of a standard cell library. Hence, providing the library that contains the delay information of every cell at the reduced voltage allows the synthesis tool to select the most suitable set of standard cells (i.e. the cells that exhibit smaller delay increase at that low voltage). Hence, circuits with smaller maximum delay can be obtained. This, in turn, can mitigate the incurred performance loss when circuits operate at NTC.

V. Parallelized Near-Threshold Computing

To compensate for the significant performance loss at NTC, more cores can be employed. Hence, designers can maximize the performance, under a specific power budget, while reducing the voltage towards NTC. However, the Amdahl serial factor (S_f) [3], which represents the obstacles that limit the parallel execution (e.g., application sequence, communication latency, synchronization, I/O etc.), needs to be considered. *Importantly, finding the OEP in which the added cores provide the maximum energy efficiency becomes more complicated due to the relation between the added cores, consumed power and overall gained performance.*

To achieve that, we, first, estimate using our voltage-aware timing and power analysis proposed in Section IV the maximum delay of the studied design and its total power at every voltage step. Then, at every voltage step, we calculate the maximum possible number of cores (n) within the provided power budget. Afterwards, we calculate the achievable performance gain from the added cores under the impact of S_f, as Eq. 3 clarifies. In this equation, the performance gain is defined as the ratio between the original performance (obtained from a single core operated at the nominal V_{dd}) and the new performance (obtained from parallelized NTC), while considering the reduction in performance, when reducing V_{dd}.

$$Performance\ Gain = \frac{S_c}{S_f + \frac{(1-S_f)}{n}} \tag{3}$$

Where S_c represents the core's performance at the low V_{dd} normalized to the core's performance before voltage scaling, i.e. at the nominal V_{dd}, e.g., 1.2V. Note that $0 \leq S_c \leq 1$.

Considering power loss at NTC: Beside its impact on the overall performance, the S_f has also an important impact of on the total consumed power. This is because the sequential portions of an application will be always executed using just a single core. Thus, during the sequential phase of computation, only one core out of n available cores will be active, whereas the remaining $(n-1)$ cores will be idle (i.e. only leakage power is then consumed). On the other hand, during the phase of parallelism all n cores will be active. Hence, when the total consumed power is calculated to check against the predetermined budget, it is essential to consider the impact of S_f. Otherwise, the total power consumption will be *conservatively* estimated assuming all cores are always active. The total power can be calculated as Eq. 4, based on [17], shows.

$$W = \frac{W_c + (n-1)W_c k_c S_f}{S_f + \frac{1-S_f}{n}} \tag{4}$$

Where W is the total power consumption normalized to the consumed power at nominal voltage, W_c is the active core's power consumption relative to the power at nominal voltage, k_c is the fraction of core's idle power normalized to the active core's power, S_f sequential fraction and n number of cores. Note that $0 \leq w_c \leq 1$ and $0 \leq k_c \leq 1$.

Algorithm 1 Finding the optimal energy point.

```
Require: Power Budget , Delay Table, Power Table, serial factor $S_f$
for each $V_{dd}$ and $S_f$ do
    Retrieve corresponding Power        ▷ based on voltage-aware Power analysis
    Retrieve corresponding Delay        ▷ based on voltage-aware timing analysis
    Find Max number of cores            ▷ based on the power budget
    Estimate Overall Performance        ▷ based on Eq. 3
    Estimate Total Power                ▷ based on Eq. 4
    Estimate Power Per Performance      ▷ based on Eq. 5
end for
Search for OEP        ▷ find where the performance or $PW$ is maximum
```

Impact of serial factor: S_f in parallelized NTC has two fold impacts: a) it leads to a loss in the performance gain that can be obtained from adding more cores, b) it causes a reduction in the total consumed power and hence the provided power budget will not be fully exploited and utilized. Unlike state-of-the-art, which neglects the impact of S_f on the total power consumption, we accurately estimate the consumed power while taking the sequential and parallel phases into consideration as Eq. 4 shows. The equation shows that during the sequential computation phase, one core in the active state consumes W_c, and all idle cores consume $(n-1)W_c k_c S_f$. During the parallel computation phase, all cores are active and they consume nW_c.

Performance Per Watt: A trade-off is essential to find OEP under conflicts (performance, power and S_f). Thus, to accurately evaluate the overall gain in parallelized NTC, it is necessary to calculate the performance per watt (PW) as Eq. 5 illustrates. PW allows designers to figure out how the power budget is *efficiently* utilized and thus at which reduced V_{dd} the efficiency of the multi-core processor is maximized.

$$PW = \frac{S_c}{W_c + (n-1)w_c k_c S_f} \tag{5}$$

Implementation: To automatically perform a design-space exploration and find where the OEP precisely occurs, we developed an automated framework that considers the interactions between the power and delay of the studied processor at every voltage step together with the required power budget and the impact of S_f. At every voltage step, the maximum number of cores in which the performance is maximized within the given power budget is calculated. Then, the optimal voltage in which the efficiency is maximized is determined. Algorithm 1 summarizes the basic functionality of our framework, where the searching complexity is $O(mk)$, where m is the number of S_f cases, and k is the number of V_{dd} scaling steps.

VI. Evaluation and Experimental Results

First, we explain our experimental setup. Then, we present the evaluation of parallelized NTC. Finally, we demonstrate the obtained speedup from our proposed voltage-aware logic synthesis for both single-core and parallelized NTC.

Experimental Setup: In our work, we focus only on the CPU and other parts like memory caches in which errors

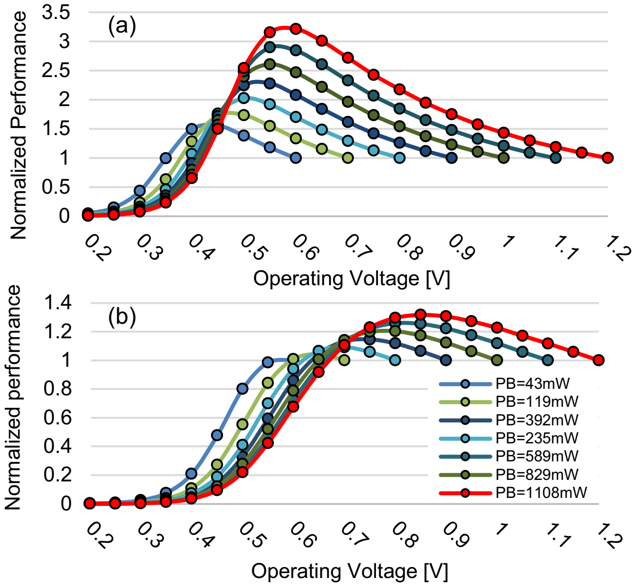

Fig. 4. Performance improvement at parallelized NTC under different power budgets for the case of S_f of (a) 0.01 and (b) 0.2. Smaller power budget pushes OEP towards the near-threshold region but when S_f is larger the OEP moves towards the super-threshold region. Note that the power budget estimated for every V_{dd}.

could occur at NTC are assumed to operate at a higher V_{dd}, as Intel proposed [18]. We consider state-of-the-art 64-bit Rocket Processor [13], which is an industry-competitive processor from Berkeley recently designed to serve for future micro-architectural. The CPU consists of 5 pipeline stages using 31 General Purpose Registers and 32 Registers for the FPU. The CPU is implemented at the 45nm technology node (details in Section IV) and the total number of sequential and combinational standard cells is around 10^6. This is around 10x larger than the largest circuit benchmark in ITC'99 benchmark circuit suite [19]. For each of logic synthesis, timing analysis, and power analysis we employed the standard tool flows provided by Synopsys.

OEP for Parallelized NTC: The impact of S_f could change OEP based on the selected optimization goal (i.e., power or performance). However, different applications with different S_f result in different behavior. Therefore, for a general analysis (i.e. considering that the application's behavior is not known in prior), we explore a wide range of S_f values in which varied runtime behaviors are reflected. In case S_f changes at runtime, designers could predict OEP from expected range of S_f and then select a conservative OEP. As explained earlier in Section V, the key focus here is obtaining at which V_{dd} the maximum energy efficiency occurs along with the maximum number of cores in which the performance is maximized while a certain power budget is fulfilled. In Fig. (4(a and b), we evaluate the achieved speedup from parallelized NTC under different power budgets at different voltages (maximum power at nominal supply voltage) in the scope of two different cases of serial factors: $S_f = 0.01$ and $S_f = 0.2$, respectively. As shown in Fig. 4(a), providing a smaller power budget pushes OEP towards the left side (i.e. to a lower V_{dd}). However, having a higher S_f has a contradictory effect as it pushes OEP towards the right side (i.e. far from NTC), as shown in Fig. 4(b) .

Figs. 5(a, b and c) explore the impact of various S_f on the gained performance as well as on shifting OEP. The evaluations have been done under one power budget which is the total consumed power of the processor at the nominal voltage of 1.2V. As expected, having a higher S_f leads to a lower speedup as Fig. 5(a) demonstrates. S_f plays a major role in determining where the OEP occurs. Higher S_f shifts OEP towards the super-threshold region. For instance, having a serial factor of merely 0.01 (i.e. only 1% of the running application is sequential and the rest can be fully parallelized) leads to shifting V_{opt} to 0.6V.

To explore further the impact of S_f, we present in Fig. 5(b) the corresponding normalized total consumed power for different S_f cases. As shown, when S_f becomes higher, the application spends more time within a single core and hence the rest of the cores are idle. Therefore, the provided power budget cannot be fully exploited. In other words, higher S_f results in less utilization of the provided power budget. Thus, evaluating the overall efficiency represented by the Performance per Watt (details in Section V) becomes essential to determine where OEP precisely occurs. As Fig. 5(c) demonstrates, the impact of S_f becomes marginal compared to the previously-observed impact on both speedup (see Figs. (5(a), 4) and utilized power budget (see Fig. 5(b)). As shown, the OEP under different S_f approximately remains within or close to the near-threshold region. *Hence, when optimizing for performance alone the OEP moves far from the near-threshold region (i.e. OEP occurs at $V_{dd} > 0.5V$). However, when optimizing for performance per watt the OEP remains close to the near-threshold region (i.e. OEP occurs at $V_{dd} < 0.5V$). Therefore, the optimization goal plays a major role in where the OEP occurs.*

Voltage-aware logic synthesis: As explained in Section IV we propose to employ our created voltage-aware cell libraries during the logic synthesis. In such a case, the mature algorithms within the synthesis tool will select the standard cells that have a better (i.e. smaller) delay at the reduced V_{dd}. Fig. 6 demonstrates (for the case of a single core) how our proposed voltage-aware logic synthesis provides a processor's netlist that exhibits a smaller delay at every V_{dd} step (i.e. higher frequency). As can be observed Fig. 6, in all V_{dd} cases, a performance improvement can be observed and it reaches around 35% when processor is operated at NTC.

Finally, we also investigate the impact of our proposed voltage-aware logic synthesis with respect to parallelized NTC. Fig. 7 demonstrates the obtained improvements with respect to performance gain at different V_{dd} under various S_f cases. As shown, our voltage-aware logic synthesis leads to above 40% higher performance between 0.25V and 0.4V (within the NTC region).

VII. Conclusion

In this work, we demonstrated how voltage-aware synthesis provides processors with a much better performance at NTC. We also showed how the serial factor in parallelized NTC plays a major role in shifting the OEP based on the selected optimization goal (performance or power optimization). We also demonstrated why voltage-aware cell libraries [4] are

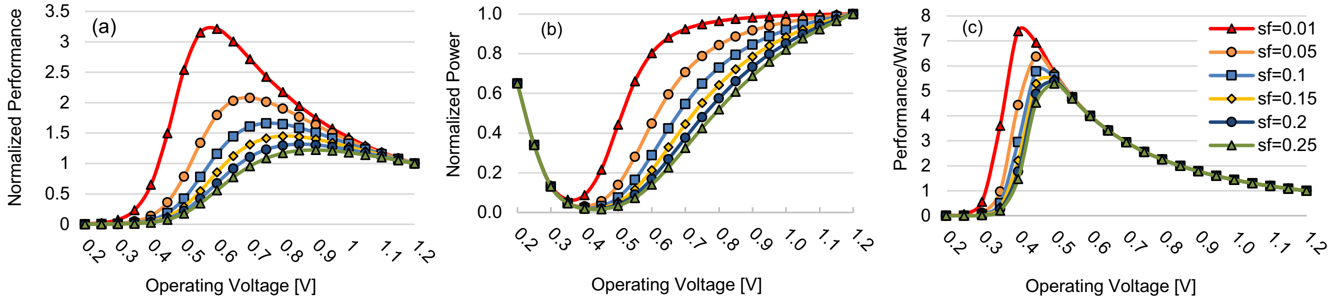

Fig. 5. Investigating the impact of serial factor on performance gain (a), power budget utilization normalized to power at nominal voltage (b) and performance per watt (c). Higher S_f results in less performance gain and less power budget utilization (see a and b). The maximum performance per watt occurs at (or close to) NTC (see c).

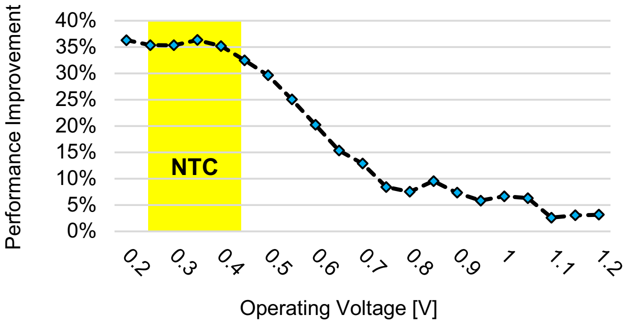

Fig. 6. Impact of our proposed voltage-aware logic synthesis on improving the performance of processor when reducing the voltage. An improvement of around 35% can be achieved at NTC.

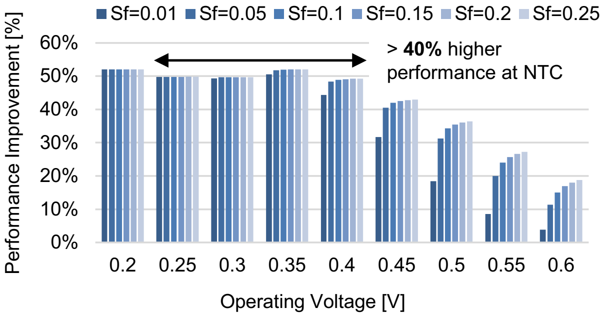

Fig. 7. Voltage-aware synthesis on improving the performance gain of parallelized NTC under different serial factor (S_f) cases.

prerequisite for efficient near-threshold computing because they allow an accurate selection of the OEP as well as performance improvements.

Acknowledgment

This work was supported in parts by the German Research Foundation (DFG) as part of the priority program "Dependable Embedded Systems" (SPP 1500 - spp1500.itec.kit.edu). We would like thank Andreas Gerstlauer for providing the RTL of DCT circuit used in the analysis in Fig. 1.

References

[1] D. Blaauw, S. Martin, T. Mudge, and K. Flautner, "Leakage current reduction in vlsi systems," *Journal of Circuits, Systems, and Computers*, vol. 11, 2002.

[2] H. Kaul, M. A. Anders, S. K. Mathew, S. K. Hsu *et al.*, "A 320 mv 56 μw 411 gops/watt ultra-low voltage motion estimation accelerator in 65 nm cmos," *IEEE Journal of Solid-State Circuits*, 2009.

[3] G. M. Amdahl, "Validity of the single processor approach to achieving large scale computing capabilities," in *Proceedings of the April 18-20, 1967, spring joint computer conference*. ACM, 1967.

[4] "Votlage-Aware Cell Libraries," http://ces.itec.kit.edu/dependable-hardware.php.

[5] V. De, S. Vangal, and R. Krishnamurthy, "Near threshold voltage (ntv) computing: Computing in the dark silicon era," *IEEE Design Test*, April 2017.

[6] M. Gautschi, P. D. Schiavone, A. Traber, I. Loi *et al.*, "Near-threshold risc-v core with dsp extensions for scalable iot endpoint devices," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2017.

[7] H. Amrouch, B. Khaleghi, A. Gerstlauer, and J. Henkel, "Reliability-aware design to suppress aging," in *Design Automation Conference (DAC), 2016 53nd ACM/EDAC/IEEE*. IEEE, 2016.

[8] H. Amrouch, B. Khaleghi, and J. Henkel, "Optimizing temperature guardbands," in *2017 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2017.

[9] N. Pinckney, L. Shifren, B. Cline, S. Sinha *et al.*, "Near-threshold computing in finfet technologies: Opportunities for improved voltage scalability," in *2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, June 2016.

[10] N. Pinckney, K. Sewell, R. G. Dreslinski, D. Fick *et al.*, "Assessing the performance limits of parallelized near-threshold computing," in *Design Automation Conference (DAC), 2012 49th ACM/EDAC/IEEE*, June 2012.

[11] Q. Xie, X. Lin, Y. Wang, S. Chen, M. J. Dousti, and M. Pedram, "Performance comparisons between 7-nm finfet and conventional bulk cmos standard cell libraries," *IEEE Transactions on Circuits and Systems II: Express Briefs*, Aug 2015.

[12] L. B. Soares, S. Bampi, A. L. R. Rosa, and E. Costa, "Near-threshold computing for very wide frequency scaling: Approximate adders to rescue performance," in *New Circuits and Systems Conference (NEWCAS), 2015 IEEE 13th International*. IEEE, 2015.

[13] "The rocket chip generator," http://www2.eecs.berkeley.edu/Pubs/TechRpts/2016/EECS-2016-17.html.

[14] "Nangate, Open Cell Library," http://www.nangate.com/.

[15] "Predictive Technology Model," http://ptm.asu.edu/.

[16] Y. Chauhan, S. Venugopalan, M. Karim, S. Khandelwal *et al.*, "BSIM - Industry standard compact MOSFET models," in *ESSCIRC*, 2012.

[17] D. H. Woo, D. H. Woo, D. H. Woo, D. H. Woo *et al.*, "Extending amdahl's law for energy-efficient computing in the many-core era," *Computer*, 2008.

[18] S. Jain, S. Khare, S. Yada, V. Ambili *et al.*, "A 280mv-to-1.2v wide-operating-range ia-32 processor in 32nm cmos," in *2012 IEEE International Solid-State Circuits Conference*, Feb 2012.

[19] F. Corno, M. S. Reorda, and G. Squillero, "Rt-level itc'99 benchmarks and first atpg results," *IEEE Design & Test of computers*, vol. 17, 2000.