MuTARe: A Multi-Target, Adaptive Reconfigurable Architecture

Marcelo Brandalero (Thesis Author)[1], **Luigi Carro (Co-Supervisor)**[1], **Antonio Carlos Schneider Beck (Main Supervisor)**[1]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{mbrandalero,carro,caco}@inf.ufrgs.br

Abstract. *With recent changes in transistor scaling trends, the design of all types of processing systems has become increasingly constrained by power consumption. At the same time, driven by the needs of fast response times, many applications are migrating from the cloud to the edge, pushing for the challenge of increasing the performance of these already power-constrained devices. The key to addressing this problem is to design application-specific processors that perfectly match the application's requirements and avoid unnecessary energy consumption. However, such dedicated platforms require significant design time and are thus unable to match the pace of fast-evolving applications that are deployed in the Internet-of-Things (IoT) every day. Motivated by the need for high energy efficiency and high flexibility in hardware platforms, this thesis paves the way to a new class of low-power adaptive processors that can achieve these goals by automatically modifying their structure at run time to match different applications' resource requirements. The proposed Multi-Target Adaptive Reconfigurable Architecture (MuTARe) is based upon a Coarse-Grained Reconfigurable Architecture (CGRA) that can transparently accelerate already-deployed applications, but incorporates novel compute paradigms such as Approximate Computing (AxC) and Near-Threshold Voltage Computing (NTC) to improve its efficiency. Compared to a traditional system of heterogeneous processing cores (similar to ARM's big.LITTLE), the base MuTARe architecture can (without any change to the existing software) improve the execution time by up to* $1.3\times$*, adapt to the same task deadline with* $1.6\times$ *smaller energy consumption or adapt to the same low energy budget with* $2.3\times$ *better performance. When extended for AxC, MuTARe's power savings can be further improved by up to* 50% *in error-tolerant applications, and when extended for NTC, MuTARe can save further* 30% *energy in memory-intensive workloads.*

1. Problem Statement and Proposed Solution

Improvements in the performance and energy consumption of hardware devices have relied for a long time on the technology scaling. In recent years, however, contrary to the scaling predictions, the benefits of shrinking technology on the power and energy consumption has decreased [Esmaeilzadeh et al. 2012]. Despite that, application innovations coming from emerging domains such as Machine Learning (ML) keep pushing the performance requirements of hardware devices [Dean et al. 2018]. To make matters

even worse, applications are slowly migrating from the cloud to the extremely resource-constrained edge devices in order to embrace the concept of fog computing and provide faster response times [Bonomi et al. 2012].

The state-of-the-art approach in achieving high best energy efficiency is to design application-specific accelerators. These hardware devices are coupled to general-purpose processors but tailored to a particular application. By doing so, they provide the exact resources that are needed for the efficient execution of the given applications [Patterson 2018]. Since these devices are fixed after manufacture, however, this approach is unable to match the pace of fast-evolving applications are deployed in the Internet-of-Things (IoT) every day. Therefore, alternative approaches are required to support the energy-efficient execution of a broad range of applications, unpredictable at design time, and with different resource requirements that may also change at run time [Adegbija et al. 2018].

Reconfigurable Accelerators (RAs) present an alternative that can address most of the issues associated with application-specific ones. Built from arrays of Processing Elements (PEs) with programmable interconnects, RAs allow customized datapaths matching each different application's needs to be defined at run time. While traditionally requiring special compilers that can generate code to reconfigure the datapaths, RAs can also be coupled with run-time code mapping techniques that enable the transparent acceleration of already-deployed code without requiring any change in the software development process [Beck et al. 2014].

Still, there is an efficiency gap between RAs and application-specific designs that must be addressed to enable the efficient execution of emerging performance-hungry applications such as those from the Machine Learning (ML) domain. To bridge this gap, this thesis presents the Multi-Target Adaptive Reconfigurable Architecture (MuTARe). MuTARe paves the way to a new class of low-power adaptive processors that can achieve high performance and high energy-efficiency while retaining their flexibility and applying to a broad range of applications. This is achieved by synergistically combining multiple adaptability techniques for a transparent matching between the hardware and the resource requirements of different applications, achieving better results than current reconfigurable designs. MuTARe is multi-target in the sense that it can be designed towards different application domains (e.g., by tuning the size of the fabric) and, besides adapting to the requirements of each application, can also adapt to different target metrics, such as a performance target (while saving the most power) or a power target (while maximizing the performance).

Fig. 1 presents an overview of the MuTARe approach. The heart of MuTARe is a parameterizable and combinatorial Coarse-Grained Reconfigurable Array (CGRA) that can be coupled to different forms of general-purpose core. These cores can have an in-order organization (for low-power domains, such as IoT), Out-Of-Order (OoO) (for high-performance domains, such as HPC), or even both of them in a big.LITTLE-like arrangement (for mobile domains requiring a broad adaptability range). The acceleration capabilities can be combined with Dynamic Voltage and Frequency Scaling (DVFS) to precisely adjust for the performance levels required, lowering the Operating Frequency (f) and Operating Voltage (Vdd) when possible to reduce power consumption. With these techniques, MuTARe can work transparently for already deployed binaries by providing,

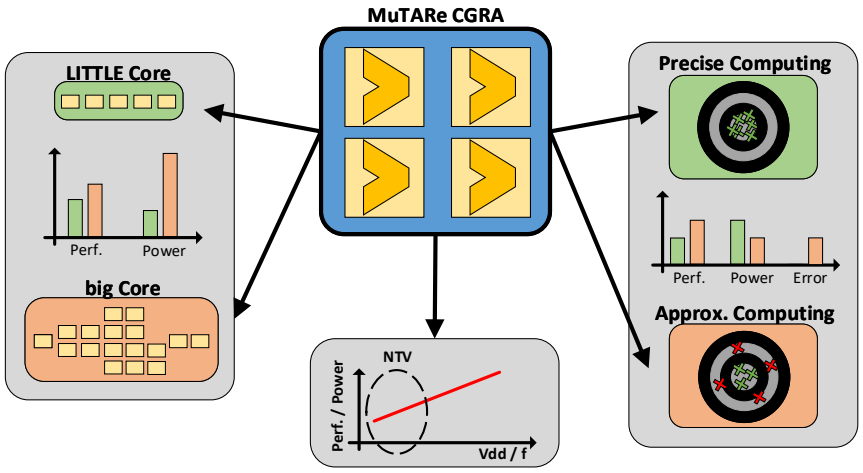

Figure 1. Overview of the proposed approach.

as a dedicated hardware module, a dynamic binary translation algorithm that automatically maps recurring instruction sequences into the CGRA for acceleration.

In the first step to move beyond traditional RAs, MuTARe provides support for drastically lowering the operating voltage beyond the traditional DVFS levels. When doing so, MuTARe can achieve the lowest energy operating point, which is typically found when transistors operate near their threshold voltage range [Mittal 2015] - Near Threshold Computing (NTC). In this challenging operating environment, however, logic elements become significantly more sensitive to fabrication variability and memories more sensitive to errors, making the reliable operation especially tricky [Rahimi et al. 2016]. MuTARe avoids this challenge by providing a suitable structure for near-threshold computing: a combinatorial CGRA which, due to its regular structure, eases the treatment of variability, and a separate higher voltage domain for the memories.

In a second step to improve over RAs, MuTARe provides support for Approximate Computing (AxC) to improve the energy efficiency in emerging error-tolerant workloads. In these workloads (the best example thereof being neural networks), intermediate computations can be carried out in an approximate manner without significantly affecting the final application output. While many works have shown the power improvements of deploying approximate functional units [Mittal 2016], general-purpose cores would be typically unsuitable for this approach since the most significant fraction of power consumption is spent in control, rather than in processing. When moving the execution from the GPP to a combinatorial CGRA, however, the potential benefits of approximate functional units can be leveraged to their full extension. Since support for approximate computing requires semantic information that must be provided by the application developer, this execution mode introduces to the base architecture Instruction-Set Extensions (ISEs) to be used towards that end.

In summary, the key contributions of this thesis are as follows:

- An adaptive microarchitecture supporting DVFS and based on a CGRA coupled to a general-purpose core that can be in-order or out-of-order (or part of a heterogeneous arrangement). Compared to existing RAs, MuTARe 1) transparently accelerates existing code, 2) can be coupled to any form of processor core, 3) can

leverage DVFS to tune the performance improvements from CGRA acceleration with the performance target and lowering the frequency if possible to save additional power. **Compared to a traditional heterogeneous Chip Multi-Processor, the base MuTARe architecture can improve the execution time by up to $1.3\times$, adapt to the same task deadline with $1.6\times$ smaller energy consumption, or adapt to the same low energy budget with $2.3\times$ better performance.**

- An adaptive microarchitecture featuring a suitable structure for implementing NTC and addressing its challenges. Compared to previous works on NTC, MuTARe first uses NTC to save power while still providing a suitable platform for executing single-threaded applications. That is achieved by compensating the performance losses from low-frequency operation by enhanced ILP exploitation, especially in memory-bound workloads, which are memory-bound. Secondly, MuTARe enables easy variability management by means of a regular fabric which can be overprovisioned with resources. **The near-threshold MuTARe implementation improves the energy consumption of memory-bound workloads by 50% with no impact on the performance**.
- An adaptive microarchitecture that can leverage the benefits of AxC and provide additional performance improvements and power savings in emerging error-tolerant domains such as ML. Compared to existing works on AxC, MuTARe 1) employs a reconfigurable accelerator is for approximate computations and 2) maintains its the general-purpose processing capabilities while leveraging the full benefits of reduced power consumption in approximate functional units, since the power consumption is switched from control in the GPP to computation in the CGRA's PEs. The result is an architecture that can be tuned at design time to better adapt for low-power or high-performance and at run-time adapt to the application being executed and improve the efficiency. **The approximate version of MuTARe can push the previously-mentioned power improvements by further 30% with minor impact on the application quality.**

2. Conclusions

This work has shown the benefits of reconfigurable acceleration for future IoT platforms that require an extensive range of adaptability. Towards this, we presented MuTARe, a hardware architecture where a single-ISA heterogeneous system is extended with a CGRA for transparent adaptability improvements and can leverage DVFS to adjust to a performance or energy constraint dynamically. Moreover, we have taken DVFS one step further into the NTV region, leveraging the structure of the CGRA to address key implementation issues of this challenging operation regime, and also use approximate computing as a further mean .

2.1. Awards

This thesis has been defended on the 18th March 2019 in the DATE conference before an international committe of experts from four different countries: Prof. Paolo Rech (UFRGS, Brazil), Prof. Cristina Silvano (Politecnico di Milano, Italy), Prof. Luca Carloni (Columbia University, USA), Prof. Michael Hübner (Brandenburg University of Technology, Germany). The thesis has received an *A* grade and honors awards. This thesis has also been awarded the *Best Thesis Award* in the *Concurso de Teses e Disertações* of *WSCAD/SBAC-PAD* (2019).

2.2. Publications

Selected Publication: The key results of this thesis have been publised on IEEE Transactions on Computers, the most prestigious journal in the field of computer organization and architecture. [doi] [Q A1].

The following works present preliminary ideas and results which were used in this thesis:

- F. M. Capella, **M. Brandalero**, L. Carro, A. C. S. Beck. *A multiple-ISA reconfigurable architecture.*. In: Springer Design Automation for Embedded Systems, v. 19, p. 329-344, 2015. [doi] [Q B1]
- **M. Brandalero**, A. C. S. Beck. *Potential of Using a Reconfigurable System on a Superscalar Core for ILP Improvements.*. In: SBESC 2014, pp. 43-48. [doi] [Q B2]
- F. M. Capella, **M. Brandalero**, J. F. Junior, A. C. S. Beck, L. Carro. *A Multiple-ISA Reconfigurable Architecture.*. In: SBESC 2013, pp. 71-76. [doi] [Q B2]

Within the scope and time frame of this thesis, the following works have been published:

- **M. Brandalero**, M. Shafique, L. Carro, A. C. S. Beck. *TransRec: Improving Adaptability in Single-ISA Heterogeneous Systems with Transparent and Reconfigurable Acceleration.*. In: DATE 2019, pp. 582-585. [doi] [Q A1]
- **M. Brandalero**, L. A. d. Silveira, J. D. Souza, A. C. S. Beck. *Accelerating error-tolerant applications with approximate function reuse.*. In: Springer Sci. Comput. Program., v. 165, p. 54-67, 2018. [doi] [Q A2]
- G. F. Oliveira, L. R. Gonçalves, **M. Brandalero**, A. C. S. Beck, L. Carro. *Employing classification-based algorithms for general-purpose approximate computing.*. In: DAC 2018, pp. 70:1-70:6. [doi] [Q A1]
- **M. Brandalero**, L. Carro, A. C. S. Beck, M. Shafique. *Approximate on-the-fly coarse-grained reconfigurable acceleration for general-purpose applications.*. In: DAC 2018, pp. 160:1-160:6. [doi] [Q A1]
- **M. Brandalero**, G. M. Malfatti, G. F. Oliveira, L. A. d. Silveira, L. R. Gonçalves, B. C. d. Silva, L. Carro, A. C. S. Beck. *Efficient Local Memory Support for Approximate Computing.*. In: SBESC 2018, pp. 122-129. [doi] [Q B2]
- **M. Brandalero**, A. C. S. Beck. *A Mechanism for energy-efficient reuse of decoding and scheduling of x86 instruction streams.*. In: DATE 2017, pp. 1468-1473. [doi] [Q A1]
- **M. Brandalero**, A. C. S. Beck. *Potential analysis of a superscalar core employing a reconfigurable array for improving instruction-level parallelism.*. In: Springer Design Autom. for Emb. Sys., v. 20, p. 155-169, 2016. [doi] [Q B1]
- L. A. d. Silveira, **M. Brandalero**, J. D. d. Souza, A. C. S. Beck. *The Potential of Accelerating Image-Processing Applications by Using Approximate Function Reuse.*. In: SBESC 2016, pp. 122-127. [doi] [Q B2]

Additionally, this thesis has also inspired the following works:

- **M. Brandalero**, T. D. Souto, L. Carro, A. C. S. Beck. *Predicting performance in multi-core systems with shared reconfigurable accelerators.*. In: Elsevier Journal of Systems Architectures, v. 98, p. 201-213, 2019. [doi] [Q B1]

- G. Korol, M. G. Jordan, **M. Brandalero**, M. B. Rutzig, A. C. S. Beck. *Power-Aware Phase Oriented Reconfigurable Architecture.*. In: ICECS 2019, pp. 626-629. [doi] [Q B1]
- G. Korol, M. G. Jordan, R. S. Silva, M. M. Pereira, **M. Brandalero**, M. B. Rutzig, A. C. S. Beck. *A Runtime Power-Aware Phase Predictor for CGRAs.*. In: ReConFig 2019, pp. 1-8. [doi] [Q A1]

2.3. Future Work

As a continuation of this thesis, the author is currently investigating the dependability aspects of the MuTARe architecture in the scope of the project *Adaptive Processor for Efficient Low-Power and Error-Resilient Execution of Emerging Embedded Applications*. The research is being conducted at the Brandenburg University of Technology Cottbus-Senftenberg (B-TU) in Cottbus, Germany, and is funded by the *Post-Doc Network Brandenburg*. This work has already resulted in the first publication, which investigates techniques for mitigating aging in CGRAs such as the one used in MuTARe:

- **M. Brandalero**, B. N. Lignati, A. C. S. Beck, M. Shafique, M. Hübner. *Proactive Aging Mitigation in CGRAs through Utilization-Aware Allocation.*. To appear in: DAC 2020. [preprint][Q A1]

References

Adegbija, T., Rogacs, A., Patel, C., and Gordon-Ross, A. (2018). Microprocessor optimizations for the internet of things: A survey. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 37(1):7–20.

Beck, A. C. S., Rutzig, M. B., and Carro, L. (2014). A transparent and adaptive reconfigurable system. *Elsevier Microprocessors and Microsystems*, 38(5):509–524.

Bonomi, F., Milito, R., Zhu, J., and Addepalli, S. (2012). Fog computing and its role in the internet of things. In *MCC Workshop on Mobile Cloud Computing*, pages 13–16.

Dean, J., Patterson, D., and Young, C. (2018). A new golden age in computer architecture: Empowering the machine-learning revolution. *IEEE Micro*, 38(2):21–29.

Esmaeilzadeh, H., Blem, E., Amant, R. S., Sankaralingam, K., and Burger, D. (2012). Dark silicon and the end of multicore scaling. *IEEE Micro*, 32(3):122–134.

Mittal, S. (2015). A Survey of Architectural Techniques for Near-Threshold Computing. *ACM Computing Surveys*, 12(4):1–26.

Mittal, S. (2016). A Survey of Techniques for Approximate Computing. *ACM Computing Surveys*, 48(4):1–33.

Patterson, D. (2018). 50 years of computer architecture: From the mainframe cpu to the domain-specific tpu and the open risc-v instruction set. In *IEEE ISSCC*, pages 27–31.

Rahimi, A., Benini, L., and Gupta, R. K. (2016). Variability Mitigation in Nanometer CMOS Integrated Systems: A Survey of Techniques From Circuits to Software. *Proceedings of the IEEE*, 104(7):1410–1448.