Variation-Aware Optimisation for Reconfigurable Cyber-Physical Systems

Rui Policarpo Duarte, Christos-Savvas Bouganis

Variation-Aware Optimisation for Reconfigurable Cyber-Physical Systems

Rui Policarpo Duarte[1], Christos-Savvas Bouganis[2]

[1] Universidade Autónoma de Lisboa, 1050-293 Lisboa, Portugal
[2] Department of Electrical and Electronic Engineering, Imperial College London, SW7 2AZ London, U.K.
rpduarte@ual.pt, ccb98@imperial.ac.uk

Abstract. Cyber-Physical Systems are present in many industries such as aerospace, automotive, health-care and transportation, and over time they have become critical and require high levels of resiliency and fault tolerance. Often they are implemented on reconfigurable logic due to IP design reutilisation, high-performance, and low-cost. Nevertheless, the continuous technology shrinking and the increasing demand for systems that operate under different power profiles with high-performance has led to implementations operating below the maximum performance offered by a particular technology. Design tools are conservative in the estimation of the maximum performance that can be achieved by a design when placed on a device, accounting for any variability in the fabrication process of the device. This work takes a new view on the performance improvement of circuit designs by pushing them into the error prone regime, as defined by the synthesis tools, and by investigating methodologies that reduce the impact of timing errors at the output of the system. In this work two novel error reduction techniques are proposed to address this problem. One is based on reduced-precision redundancy and the other on an error optimisation framework that uses information from a prior characterisation of the device. Both of these methods allow to achieve graceful degradation in performance whilst variation increases.

Keywords: FPGA, error minimisation, over-clocking, reduced-precision redundancy, Bayesian optimisation

1 Introduction

The constant fabrication process scaling has led to devices operating faster, consuming less power, but with increased variability in their fabrication. Hence, transistors are not created equally on the same device. This negatively impacts the performance of the device. Besides variations introduced by the physical constraints, transistors are also affected by other parameters such as voltage and temperature. Therefore, the newly fabricated devices are more susceptible to such variations [1] as the technology is further

reduced. Modern devices are limited by their worst performing transistor for a given family of devices.

To guarantee error-free operation of the implemented designs, once synthesised, the synthesis tools are prudent in determining the maximum clock frequency of digital circuits for a set of devices to operate on an error-free regime. As such, there is a significant gap between the maximum clock frequency reported by the models used in synthesis tools, and the actual maximum clock frequency that the actual device can operate, where the circuit will be placed on.

This work investigates mechanisms to increase the throughput of arithmetic units on Field-Programmable Gate Arrays (FPGAs) under Process-Voltage-Temperature (PVT) variation without changing the algorithm being implemented, while investigating the tradeoff in throughput, circuit area and timing errors.

Many times Cyber-Physical Systems (CPSs) are implemented on FPGAs because of the advantages they offer as high-performance, low-power, and highly specialised embedded blocks. Moreover, FPGAs are well positioned to tackle the aforementioned research problems because of their reconfigurability capabilities which is essential for the characterisation process that no other competitive technologies offer.

Examples of candidate applications for such CPSs that demand real-time performance, under different operating conditions (i.e. high-performance and low-power) are: Synthetic Aperture Radar (SAR) [2] for real-time, or high-performance; and medical capsule robots [3], and Electroencephalogram (EEG) [4] for low-power.

2 Contribution to Cyber-Physical Systems

CPSs can be seen as the combination of control theory and computer engineering, leading to control and computing co-design. They are usually comprised of sensors, actuators, and (feedback) controllers. The complexity of modern CPSs requires the usage of computational platforms to process large volumes of data. Nevertheless, these systems are discrete and suffer from delays (constant and variable) and also in variations in the system's response, which can make them instable. Moreover, data dependent control systems execute different branches of software, inducing different responses by the system. To minimise the influence of software, the most sensitive parts of these systems are implemented in hardware, usually in computational data-paths for increased performance and reduced delays. Hence, real-time CPSs use dedicated hardware (co-)processors in reconfigurable logic which ensure high-performance, constant throughput and flexibility to instantiate customised sub-systems. Nevertheless, as any other silicon device, reconfigurable logic is sensitive to degradation, and variation of its operating conditions. Consequentially, the work here presented makes important contributions to the implementation of more resilient CPSs on reconfigurable devices, by promoting their graceful degradation.

In a nutshell, this research contributes to the advancement of CPSs by investigating alternatives to reduce, or mitigate, timing errors in applications that tolerate some errors in its calculations, and proposes methods to close an existing gap in this research area. Additionally, a possible by-product of this research is the promotion of security in circuit designs by turning the replication of the CPS unfeasible (i.e. Physical Unclonable Functions).

3 Background and Related Work

3.1 Sources of Variation

The performance of transistors is affected by variation in their sizes, supply voltages, temperature, cross-talk and jitter [7]. Contributions [5,6] present how the performance of devices from the same family degrades with the aging of the device.

3.2 Variation-Aware Methods for Throughput Increase in FPGAs

Several methods have been proposed to minimise, or recover from, timing errors caused by the aforesaid sources of variation. They operate on different levels of the design, namely placement and routing, Register-Transfer Level (RTL) and algorithm level.

Variation-aware placement and routing [8] makes use of a model created from a characterisation of the fabric where the design is going to be deployed. The main benefit of this method is to instruct the placement and routing tool [9] to assign the critical-paths to the fastest elements on the device, thus reducing the critical-path delay, increasing the designs' overall clock frequency. The drawback is the necessity to characterise the design before deploying the design, which is currently only available on FPGAs [10].

In order to increase the clock frequency of a datapath, on the RTL level, typically the Digital Signal Processing (DSP) designer either reduces its word length or introduces additional pipeline levels. Word length optimisation while offering reductions in area and delay, it also reduces the precision of the results produced. Extensive work has been published on the aforementioned topics, offering techniques to implement and optimise DSP designs [11,12,13].

3.3 Error Recovery Methods

Since the early days of computing, engineers have been concerned with faults and errors from different natures [14]. Most of the mechanisms and methodologies proposed to mitigate them rely on extra circuitry and processing time. Usually a compromise is achieved in terms of the minimum requirements by the application to work, amount of

resources and the time to produce results. Recently [15] has proposed a method to adapt the circuit's voltage according to the level of errors, thus trading off power for accuracy. However, some circuits don't admit errors of any kind in their computations. Hence, circuits that require a deterministic output in their calculations have to rely on other methodologies to recover from errors [16], such as Razor [17]. Other alternatives for fault tolerance techniques, their benefits and their limitations have been presented in [18]. Reduced-Precision Redundancy RPR was originally presented in [19] as a mechanism to contain errors in designs under voltage over-scaling, for low-power, based on the assumption that DSP design can tolerate some errors in their calculations, trading off precision for power [20,21].

It relies on a smaller implementation of the original system computed in parallel using truncated operands. It compares both outputs and verifies if the magnitude of the difference is below a user defined limit. The method selects the original output if this result is below the specified threshold (T), and the approximated result otherwise. Reduced-Precision Redundancy (RPR) has been compared with other error recovering schemes, such as Triple-Modular Redundancy (TMR) [22], in terms of the tradeoff between errors and resources.

3.4 Resource Optimisation through Bayesian Inference

A method to optimise linear projection implementations through inference was first introduced in [23] and later extended in [24,25]. Here, the problem is to discover a projection matrix to compute the best approximation for the original data, minimising resources and reconstruction Mean-Square Error (MSE) of the projected data in the original space. One of the improvements, compared to other works is the avoidance of exhaustive search for solutions.

Being the factors F from a linear projection of data X and the basis matrix Λ, searching for a possible solution for Λ and F is an ill-conditioned problem, for which solutions from heuristic methods are suboptimal. The framework [25] uses a Bayesian conception of the factor analysis model rather than the Karhunen-Loeve Transformation (KLT) algorithm to find the elements of the Λ matrix that minimise the area cost, rather than a constant area cost consideration in the KLT algorithm. The framework receives the problem data, the area models and its parameters and iteratively computes the basis for all projection vectors. As a result it returns the basis matrix and the assignment for the logic elements.

3.5 Summary

Here are some of the most important concepts present in the design of DSP designs for FPGAs focused on performance optimisation, and also methods to mitigate timing errors, or to achieve graceful degradation.

Table 1 summarises the different techniques that can be applied to minimise, or minimise, timing errors. For each technique it shows how it actuates and its limitations.

Table 1. Summary of the existing techniques for acceleration of computations, in order to mitigate timing errors, and their limitations.

Technique	Effect	Limitations
Deep pipelining	Break the critical-path by inserting registers	Unsuitable for some streaming algorithms, or algorithms using recursion
Word length optimisation	Reduce the critical path by processing less bits from truncated operands	Penalty in the quality of results
Razor [17]	Check if the output matches the shadow register	Temporal redundancy is unsuitable for streaming applications
Reduced- Precision Redundancy [19]	Check if the error is within a threshold	Requires extra latency. Unsuitable for algorithms using recursion

4 Reduced Precision Redundancy Framework

RPR as originally proposed in [19] replaces the result produced by the arithmetic unit with an approximate result, which is computed in parallel, in order to control the propagation of errors. This tool operates similarly as existing RPR implementations as it wraps the original combinatorial unit with a redundant circuitry. Yet, it separates itself from other architectures as it has characteristics that are distinct for high-throughput and applications intolerant to latency, and depends on a novel RPR architecture with zero latency cost.

This tool reuses the concept of substituting a set of Most Significant Bytes (MSBs) from the original arithmetic unit with an approximate result, in case of error, nevertheless it introduces new mechanisms to: a) identify timing errors and b) generate approximations. Typically, RPR architectures use some of the MSBs of arithmetic operators, as usually they are the critical paths.

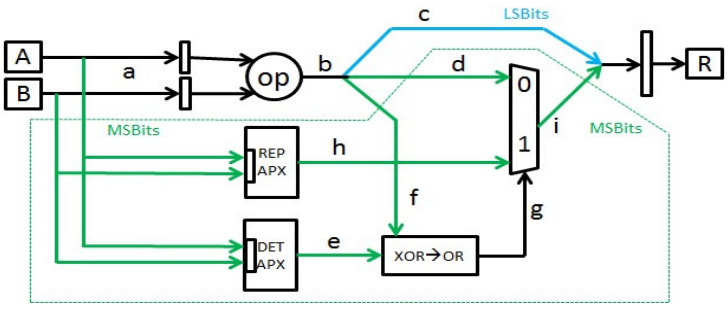

Fig.1. The new RPR architecture on a generic operator.

4.1 Architecture

The two main novelties in this architecture are: a) the use ROMs to have the MSBs of the approximations, rather than the simplified operator; and b) a bit-wise testing (XOR→OR) as a substitute of a subtraction before the comparison with a user-defined limit to identify errors in computations.

Fig. 1 presents the new architecture in a combinatorial unit op. Its inputs are *A*, *B*; and *R* is its output. Label *a* represents the inputs and *b* the original operator's output. The other labels refer to the paths added by the RPR. They are explained in more detail in [26].

The use of ROMs and bit-wise output testing (XOR→OR) minimises the delay between the input output ports of the new RPR unit. As long as the difference between the delay of the approximation's output and the original result inflates, it favours the increasing of the clock frequency.

As a results of potential realizations with the new RPR scheme, a taxonomy is presented to support the identification of key elements' word-lengths, such as: original unit, error-detection and error-correction ROMs. The structure is given by: iWL and oWL as the input and output word-lengths, and Ori, Det and Rep depict the Original, Detection and Replacement correspondingly: Ori iWL : Ori oWL / Det iWL : Det oWL / Rep iWL : Rep oWL

Moreover, to discriminate the many RPR architectures, the succeeding prefix is included to the previous taxonomy, which is followed hereafter:

- LUT-SUB - existing RPR but without registers; approximate result is computed with a truncated arithmetic unit implemented with Look-Up Tables (LUTs)/ Logic Elements (LEs); errors are identified by the absolute difference between the output of the original unit and the approximation;
- ROM-XOR - proposed RPR; detection approximation is obtained from a ROM, and errors are detected via a bitwise testing of the MSBs with the original unit.

4.2 Approximation Functions

The approximation functions aim to precisely produce the original unit's MSBs in order to decrease the approximation errors. The novelty when compared to other variations is the opportunity to use any approximation function, rather than depending on a truncated arithmetic unit computed simultaneously.

The new architecture employs two approximations simultaneously. One approximation in error detection (*DET APX*) combined with another approximation for replacement (*REP APX*) of erroneous results. Since data is removed from an approximation computed via truncated input arguments, a bit-wise testing between the expected and approximation results will be identified as a mismatch. Hence, *REP APX* is used to correct those MSBs.

4.3 ROM-XOR RPR Arithmetic Operators

The tool here presented supports any arithmetic unit. The details about how they are supported by this tool can be found in [26]. The tool is indifferent to synthesis of the operator and can be adapted to any arithmetic operator of any word-length.

Adder In this operator the linear approximation function examined was: $apx(a,b,k) = a+b+k$. Here a and b are the truncated input operands and k is an offset to balance the truncation of the input operands. The new RPR tool searches for values of k which minimise the objective function regarding the predicted and the approximation MSBs.

Multiplier The RPR multiplier is similar to the RPR adder, being the arithmetic unit the only change. In this case, a new approximation function is investigated to attenuate the objective function. The approximation function is given by: $apx(a,b,m,l) = (a + m) * (b + m) + l$. Here a and b are the truncated operands, and m and l are the approximation function coefficients.

4.4 Performance Evaluation

To assess this work, it was compared against standard arithmetic units, with and without RPR. It employs the default implementation for the synthesis of the combinatorial operators.

To obtain precise results, the device was kept at constant temperature of 20 degrees Celsius, by using a Peltier element on top of the FPGA device and a 1,2V core voltage from an independent power source. All tests were conducted on a EP3C16F484C6 Cyclone III FPGA from Altera [27], on a DE0 board from Terasic [28].

Multiplier Three 8-bit multipliers were used in the performance comparison: no RPR, LUT-SUB RPR and ROM-XOR RPR. Fig. 2 presents the variance and mean of the difference between the expected value and the value obtained from the board for all

multipliers. The top clock frequency of the ROM-XOR-RPR scheme is near the clock frequency of the multiplier without redundancy, and it surpasses any other multiplier being extremely over-clocked, e.g. 300 MHz. Going beyond 340MHz the ROM-XOR-RPR multiplier has similar variance values as the other multipliers. Nevertheless, its mean error is still near zero.

Linear Projection Designs Karhunen-Loeve transformation (KLT) [29], or linear projection, is normally utilised to compress data. In this assessment, a 8:16/5:3/5:3 ROM-XOR RPR, a LUTSUB RPR with an approximation obtained from the 5 MSBs and a threshold equivalent to the 2 MSBs, is compared against the implementation of the linear projection without any redundancy.

The circuit to compute the linear projection corresponds to a fused Multiply-Accumulate (MAC) for each projected dimension. In the implementation all inputs are encoded with 9-bit sign-magnitude. The output of the multiplier is 16 bits unsigned. The word-length of the output increases with the number of accumulation stages (log_2).

Fig. 3 presents the results for the 3 implementations at 270MHz. The top row corresponds to the anticipated error-free result. The rows below relate to the results for: NO RPR, LUT-SUB RPR and ROM-XOR RPR. For all images the PSNR is computed from the projected data reconstruction, on the FPGA, into the original space in software. It's clear that the ROM-XOR RPR provides linear projections circuits with less errors and produce the smallest reconstruction Peak Signal-to-Noise Ratio (PSNR) for all images.

The results demonstrate that a small impact in the top clock frequency in the error-free regime is paid back when in the error prone regime.

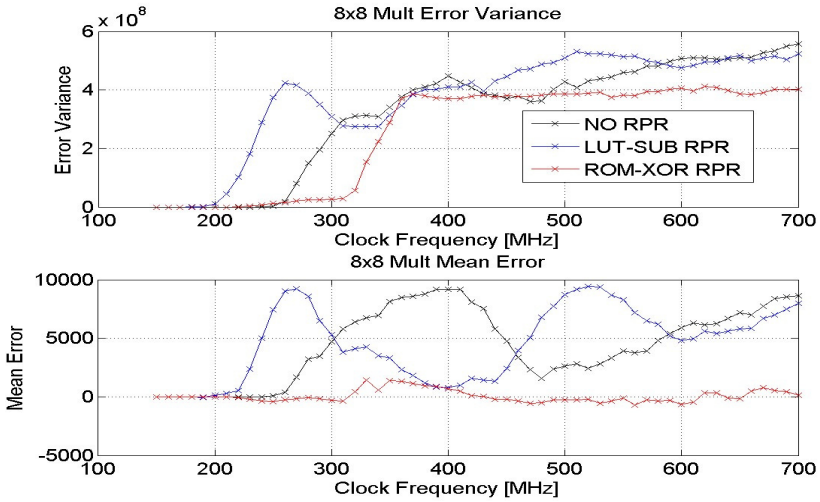

Fig.2. Error variance (top) and mean error (bottom) of an 8-bit unsigned LUT-based multipliers for different clock frequencies.

5 Optimisation of Linear Projection Designs for Over-Clocking

In the circuit to implement the Linear Projection, or KLT, design, the data path holds the most critical paths. The main focus of this work is on over-clocking multiplier circuits, as they're the components with the largest delay in the data path of the design. In the KLT algorithm, the calculation of the projection matrix Λ and its hardware mapping onto FPGAs are often considered as two independent steps in the design process. However, considerable area savings can be achieved by coupling these two steps as shown in [23,25]. The Bayesian formulation presented considers the subspace estimation and the hardware implementation simultaneously, allowing the framework to efficiently explore the possibilities of custom design offered by FPGAs. This framework generates Linear Projection designs which minimise errors and circuit resources, when compared to the standard approach of the KLT transform application followed by the mapping to the FPGA.

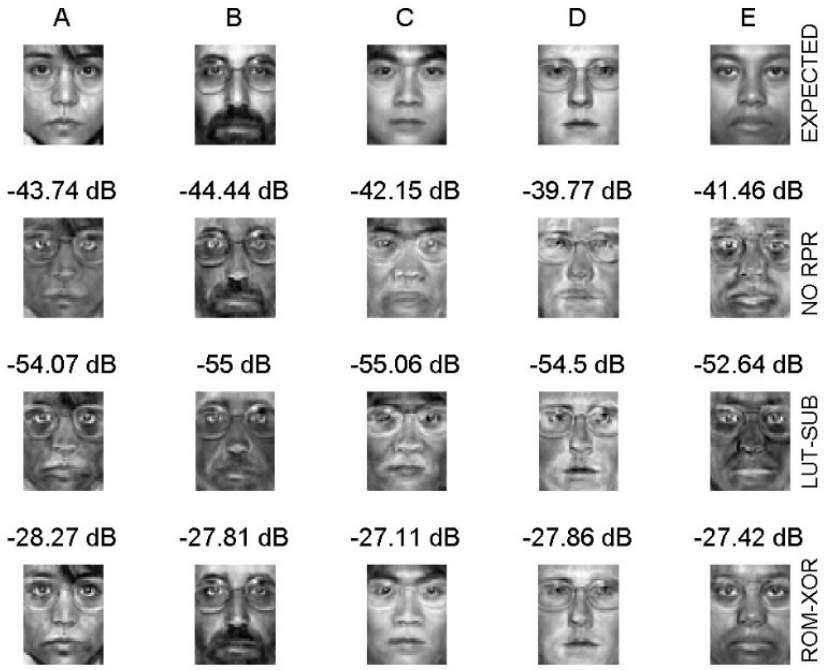

Fig.3. Images of reconstructed faces (A-E) in the original space without timing errors (EXPECTED), obtained from different multiplier implementations (NO RPR, LUT-SUB RPR, ROM-XOR RPR) at 270 MHz. On top of each face there's the corresponding reconstruction error.

A key idea from [25] is to inject information about the hardware (i.e. in this case about the required hardware resources of a Constant Coefficient Multiplier (CCM)) as a prior knowledge in the Bayesian formulation of the above optimisation problem. In more detail, the proposed framework in [25] estimates the basis matrix Λ, the noise covariance Ψ, and the factors using Gibbs sampling algorithm [30] from the posterior distribution of the variables, having injecting knowledge about the required hardware recourses for the implementation of the CCMs through a prior distribution. Thus, a probability density function is generated for the unknown Λ matrix, which is used to for generation of samples, where the prior distribution tunes this posterior distribution, and thus accommodating the impact the required hardware resources. [31,32,33,34] provide an extension of the above work for the optimisation of Linear Projection designs using different arithmetic units implementations, to combat the effects of circuit area, performance variation and error minimisation.

The proposed work aims to support other arithmetic unit architectures and PVT variation in the characterisation, error modelling, and generation of designs to implement. The framework selects the multipliers used for the implementation of each dot product in along with the coefficients of the Λ matrix that define the lower dimension space.

5.1 Defining the Objective Function

In this work, the objective function is based on the Mean Square Error (MSE) of the reconstructed data when are projected back to the original space, as well as on arithmetic errors at the output of the embedded multipliers that are generated when the design is over-clocked, due to PVT variation. In order to simplify the optimisation process and avoiding the formation of a multi-objective function, in this work it is assumed that the errors at the output of the embedded multipliers are uncorrelated for consecutive inputs. As such, the objective function is formed only by the errors due to dimensionality reduction and the variation of the error at the output of the multiplier when it is stimulated by a random input.

Error Models The proposed framework builds a database of the errors that can be observed at the output of the multipliers when one of the multiplicands is fixed, modelling the constant coefficient of the Λ matrix. The process is repeated for a set of frequencies, as well as for a set of multiplicands resembling in this way the possible values of the coefficients in the the Λ matrix. The observed error variance at the output of the multiplier under the different operating conditions can be seen as the uncertainty in the computations which needs to be minimized.

Prior Distribution The prior distribution $p(\cdot)$ in the framework aims to favour designs that are known (due to the previous error profiling) to perform poor under certain conditions. As the expectation of the error at the output of an overclocked multiplier can be compensated, the focus is on the minimisation of the variance of the error, which in effect it resembles the uncertainty in the computation. Furthermore, the prior distribution

can also favour certain coefficients in terms of how well the corresponding constant coefficient multiplier "fits" in the FPGA device in terms of resources, power, and so on. However, in this work, the above has not been taken into account and thus an informative prior on the value of the coefficients has been employed. Thus, the employed prior distribution captures only information regarding the errors at the output of the embedded multipliers, as a function of the targeted clock frequency, the actual physical placement on the FPGA device, the utilised core voltage, and finally the operating temperature of the device.

5.2 Design Exploration

The optimisation problem falls under Bayesian Inference, where the aim is to infer the distribution of the parameters (i.e. the coefficients of the Λ matrix) that would approximate (in MSE manner) the targeted data. In this work we selected to use Gibbs sampling [30], a sampling methodology that breaks the joint distribution to conditional distributions and samples the parameters one at a time, leading to computational efficient implementations. The sampled design points (effectively the coefficients of the Λ matrix) converge to designs that minimise the objective function U. To further improve the computational complexity, the proposed framework samples each dimension (i.e. column) of the Λ matrix sequentially. The designer provides the dimensionality of the low-dimensional space K, as well as the targeted operating conditions of the device, which define the prior distribution $p(\cdot)$ and effectively guide the framework to select coefficients that perform well under the targeted operating conditions.

5.3 Performance Evaluation

The proposed framework is evaluated for a set of Linear Projection problems when various operating conditions such as process variation, voltage, and temperature are targeted. The performance of the resulting design from the proposed framework was compared to standard design (i.e. baseline) that follows the utilises the KLT algorithm for the computation of the coefficients for the Λ matrix, thus it is unaware of the actual operating conditions and the variance of the error that is expected at the output of the multipliers. The DEO board from Terasic was used for the experiments, which hosts a Cyclone III EP3C16 FPGA. The control of the core voltage that was supplied to the FPGA was done through the PL303QMD-P [35] power supply from TTI. To fine control the temperature of the device, a thermoelectric cooler was placed on top of the device and it was calibrated using a digital thermometer from Lascar Electronics [36] with a deviation below 1◦C. It should be noted that all the reported results have been collected by running the actual system.

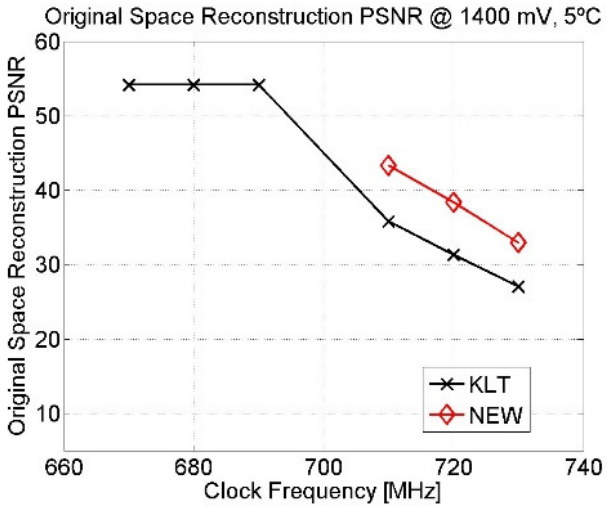

Fig.4. Performance comparison between the design generated by the proposed framework (NEW) and the reference design (KLT) for a number of operating frequencies when the supply voltage is1400mV and the device temperature is kept at 5◦C.

As a case study the problem of generating a Linear Projection design that projects data from Z^6 to Z^3 is utilised. The characterisation of the embedded multipliers under various conditions as well as the estimation of the projection matrix was performed utilising a different set of data to the ones that were used of the computation of the reconstruction error of the system (i.e. evaluation of the design). The utilised metric for the performance of a design is the PSNR of the reconstructed data in the original space.

Optimisation for Maximum Performance This scenario captures the case where maximum performance is required from the system, which requires an increased FPGA core voltage and the application of an active cooling method for the device. In our test, the device was kept at 5°C and supplied with 1400mV, instead of the 1200mV specified as the maximum supply voltage by the manufacturer. Further than that, the device was clocked with a clock frequency that was double the the maximum frequency specified by the synthesis tool for the normal working conditions. Figure 4 shows the obtained results for a number of frequencies. The results demonstrate that the proposed methodology can generate designs that result in a gain of 10dB in PSRN of the reconstruction compared to the baseline design utilising the KLT algorithm. Moreover, fixing the targeted PSNR, the designs generated by the proposed framework can be clocked 20MHz higher than the designed based on KLT.

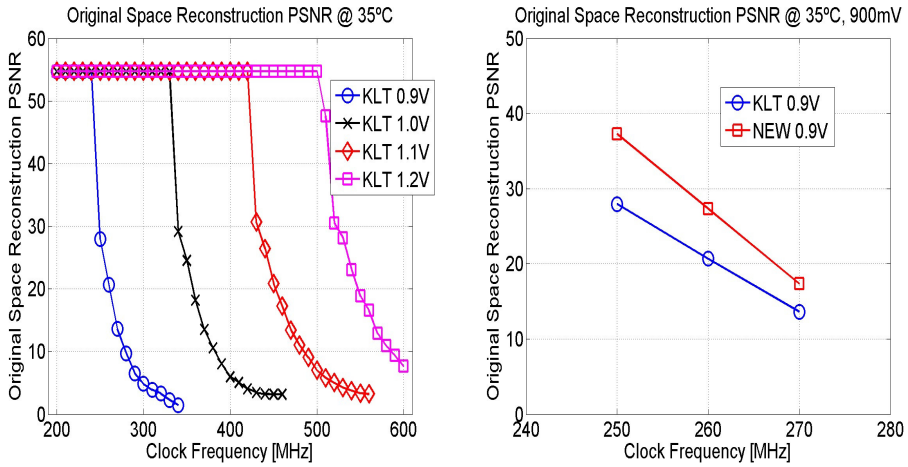

Fig.5. Performance of the KLT Linear Projection application under different core voltages (left), and a comparison between the two methods at 900mV (right).

Optimisation for Low Voltage This case investigates the gains providing by the proposed framework when a low-power system is targeted, by utilising a low core voltage and without any active cooling component. Figure 5 (left) demonstrates the performance achieved by the reference design that utilises the KLT algorithm when the device operates at 35°C under a set of FPGA core voltages (0.9V to 1.2V) and a number of clock frequencies. The figure shows that as the core voltage drops, the maximum frequency where the design operates without computational errors decreases. Figure 5 (right) focuses on a subset of these results, and in particular the design point with core voltage of 900mV. The figure depicts also the performance achieved by the design generated by the proposed framework. The results show that an improved PSNR (around 10dB) is achieved by the design generated by the proposed framework, compared to the baseline design for the same clock frequency. Moreover, for a similar PSNR, a higher clock frequency of up to 10MHz is achieved by the new design compared to the reference design.

Optimisation for Device Temperature Tolerance This test scenario investigates the case where the generated design operates under various temperature conditions. A common design methodology to address the problem is to design for the worst case condition. The proposed framework has been extended in order to address the following case in order to generate designs that would achieve a good performance under various temperatures.

In this work, in order to capture the performance of the multiplier under a set of temperatures, a weighted average of the characterisation errors was utilised. In this test case, the design is assumed to operate in temperatures: 20, 35 and 50°C with a

proportional time spend in each temperature captured by the following weights: $\alpha_{20} = 0.3$, $\alpha_{35} = 0.5$ and $\alpha_{50} = 0.2$. Please note that these weights are also used in the weighted average of the characterisation errors. In practice, the proposed framework generates circuit designs per clock frequency, covering all the temperatures within the expected range. They are identified with **NEW WAVG** in the results.

Figure 6 (top-left) depicts the performance of the reference Linear Projection circuit (KLT) as a function of the utilised frequency for the targeted temperatures of the device, when the core voltage is 1200mV. The rest of Figure 6 focuses on the comparison between the reference (KLT design), the design produced by the proposed framework using the average weight approach (NEW WAVG), and the design generated by the proposed framework assuming that highest operating temperature (i.e. worst case scenario), for a set of frequencies and temperatures.

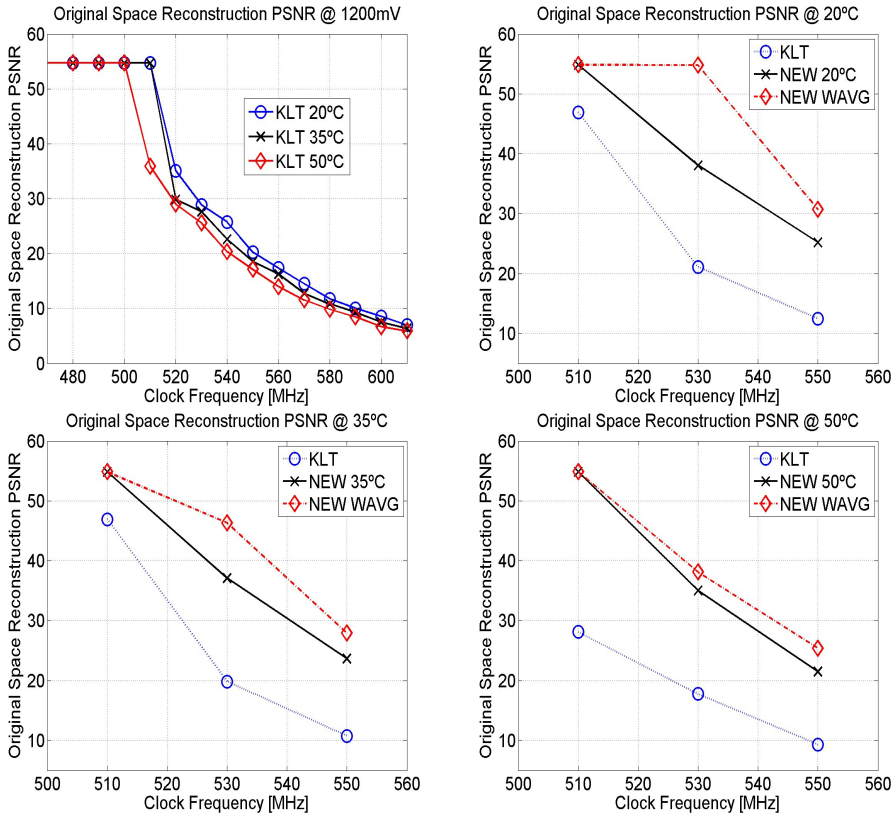

Fig.6. Performance of the Linear Projection application for a set of device temperatures (top-left), and a comparison between the three methods at 20◦C (top-right), 35◦C (bottom-left) and 50◦C

(bottom-right).

The figure shows that the designs generated by the proposed framework outperform the
Designs based on KLT across all operating temperatures and frequencies, providing at the same time significant improvements on the reconstruction of the data. Moreover, the NEW WAVG designs perform significantly better than the NEW ones for the two out of the tree operating temperatures, as they incorporate information about the performance of the device at these temperatures, performing slightly worst than NEW at 50°C, as NEW has specifically optimised for this temperature.

6 Conclusions

The constant scaling in the fabrication process has led to devices exhibiting an increase in their process variation. Hence, when the maximum throughput offered by traditional design techniques isn't enough, over-clocking the design is a method to increase it. However, this makes the design susceptible to produce errors. This work investigated methods to assess the impact of errors on arithmetic units and applications, on devices under variation, as well as methods to mitigate those errors.

The proposed RPR scheme fulfils the need for a generic method that can provide resilience to a data path without introducing extra latency, neither having to change the implementation of the algorithm. The (non-trivial) solution imagined proved to work by accelerating the units in the data path while controlling the errors. Despite the fact that the novel architecture requires twice the LEs and 2 ROMs, tests have showed that the quality of the results at the output of the RPR unit can't be achieved by other mitigation methods for the same operating conditions. Moreover, even though only timing errors were considered in this work, it can be utilised to mitigate permanent faults.

In scenarios where resources are scarce and hence it's not feasible to add extra resources to mitigate errors, the optimisation framework uses information from the previous characterisation to create an error model. From that model it generates linear projection designs, through an inference method that can produce results with less errors, when compared to traditional implementations operating under the same conditions. This method is suitable to be adopted in FPGAs, due to its reconfigurability properties, as it allows to have a prior characterisation and a later implementation on the same device. It was also identified that when accounting for timing errors, throughput, errors and area, the designs generated by the optimisation framework were the ones offering the best trade-off.

References

1. P. Sedcole and P. Y. K. Cheung, "Parametric yield modeling and simulations of FPGA circuits considering within-die delay variations," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 1, pp. 10:1–10:28, June 2008.
2. J. Nascimento and J. Bioucas Dias, "Vertex component analysis: a fast algorithm to unmix hyperspectral data," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 43, pp. 898–910, April 2005.
3. M. Beccani, H. Tunc, A. Taddese, E. Susilo, P. Volgyesi, A. Ledeczi, and P. Valdastri, "Systematic design of medical capsule robots," *Design Test, IEEE*, vol. 32 , pp. 98–108, Oct 2015.
4. L. Ke and R. Li, "Classification of eeg signals by multi-scale filtering and pca," in *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, vol. 1, pp. 362–366, Nov 2009.
5. E. A. Stott, J. S. Wong, N. P. Sedcole, and P. Y. K. Cheung, "Degradation in FPGAs: measurement and modelling," in *FPGA*, pp. 229–238, 2010.
6. J. S. J. Wong, P. Sedcole, and P. Y. K. Cheung, "Self-measurement of combinatorial circuit delays in FPGAs," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 2 , pp. 10:1–10:22, June 2009.
7. P. Sedcole, J. S. Wong, and P. Y. K. Cheung, "Characterisation of FPGA clock variability," in *Proc. IEEE Computer Society Annual Symp. VLSI ISVLSI '08*, pp. 322–328, 2008.
8. Z. Guan, J. Wong, S. Chaudhuri, G. Constantinides, and P. Cheung, "A two-stage variation-aware placement method for fpgas exploiting variation maps classification," in *Field Programmable Logic and Applications (FPL), 2012 22nd International Conference on*, pp. 519–522, Aug 2012.
9. V. Betz and J. Rose, "VPR: A new packing, placement and routing tool for FPGA research," in *Field-Programmable Logic and Applications*, pp. 213–222, 1997.
10. J. S. J. Wong and P. Y. K. Cheung, "Timing measurement platform for arbitrary black-box circuits based on transition probability," 2013.
11. G. A. Constantinides, P. Y. K. Cheung, and W. Luk, *Synthesis and optimization of DSP algorithms*. Norwell, MA, USA: Kluwer Academic Publishers, 2004.
12. J. Deschamps, G. Bioul, and G. Sutter, *Synthesis of arithmetic circuits: FPGA, ASIC, and embedded systems*. John Wiley, 2006.
13. R. E. Moore, "Automatic error analysis in digital computation," Technical Report Space Div. Report LMSD84821, Lockheed Missiles and Space Co., Sunnyvale, CA, USA, 1959.
14. J. Neumann, "Probabilistic logics and the synthesis of reliable organisms from unreliable components," 1956.
15. P. K. Krause and I. Polian, "Adaptive voltage over-scaling for resilient applications," in *Proc. Design, Automation & Test in Europe Conf. & Exhibition (DATE)*, pp. 1–6, 2011.
16. D. Roberts, T. Austin, D. Blauww, T. Mudge, and K. Flautner, "Error analysis for the support of robust voltage scaling," in *Proc. Sixth Int. Symp. Quality of Electronic Design ISQED 2005*, pp. 65–70, 2005.

17. C. Ziesler, D. Blaauw, T. Austin, K. Flautner, and T. Mudge, "Razor: A low-power pipeline based on circuit-level timing speculation," 2003.
18. U. Sharma, "Fault tolerant techniques for reconfigurable platforms," in *Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India*, A2CWiC '10, (New York, NY, USA), pp. 60:1–60:4, ACM, 2010.
19. B. Shim and N. Shanbhag, "Reduced precision redundancy for low-power digital filtering," in *Signals, Systems and Computers, 2001. Conference Record of the Thirty-Fifth Asilomar Conference on*, vol. 1, pp. 148–152 vol.1, 2001.
20. R. Hegde and N. R. Shanbhag, "Energy-efficient signal processing via algorithmic noise-tolerance," in *Proceedings of the 1999 International Symposium on Low Power Electronics and Design*, ISLPED '99, (New York, NY, USA), pp. 30 –35, ACM, 1999.
21. J. Huang, J. Lach, and G. Robins, "A methodology for energy-quality tradeoff using imprecise hardware," *DAC*, 2012.
22. R. Hentschke, F. Marques, F. Lima, L. Carro, A. Susin, and R. Reis, "Analyzing area and performance penalty of protecting different digital modules with Hamming code and triple modular redundancy," in *Proceedings of the 15th symposium on Integrated circuits and systems design*, (Washington, DC, USA), pp. 95–, IEEE Computer Society, 2002.
23. C. S. Bouganis, I. Pournara, and P. Y. K. Cheung, "Efficient mapping of dimensionality reduction designs onto heterogeneous FPGAs," in *Proc. 15th Annual IEEE Symp. Field-Programmable Custom Computing Machines FCCM 2007*, pp. 141 – 150, 2007.
24. C.-S. Bouganis, S.-B. Park, G. A. Constantinides, and P. Y. K. Cheung, "Synthesis and optimization of 2D filter designs for heterogeneous FPGAs," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 1, pp. 24:1–24:28, January 2009.
25. C.-S. Bouganis, I. Pournara, and P. Cheung, "Exploration of heterogeneous FPGAs for mapping linear projection designs," vol. 18, no. 3, pp. 436–449, 2010.
26. R. Duarte and C.-S. Bouganis, "Zero-latency datapath error correction framework for over-clocking DSP applications on FPGAs," in *ReConFigurable Computing and FPGAs (ReConFig), 2014 International Conference on*, pp. 1–7, Dec 2014.
27. Altera, "Cyclone III device handbook." Online. http://www.altera.co.uk/.
28. Terasic Technologies, "Terasic DE0 board user manual v. 1.3," 2009.
29. H. Hotelling, "Analysis of a complex of statistical variables into principal components," *Journal of Educational Psychology*, vol. 24, pp. 417–441, 1933.
30. S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-6, pp. 721 –741, nov. 1984.
31. R. Duarte and C. Bouganis, "High-level linear projection circuit design optimization framework for FPGAs under over-clocking," in *Field Programmable Logic and Applications (FPL), 2012 22nd International Conference on*, pp. 723–726, Aug 2012.
32. R. P. Duarte and C.-S. Bouganis, "A unified framework for over-clocking linear projections on FPGAs under PVT variation," in *Applied Reconfigurable Computing (ARC), 2014 10th International Symposium on*, pp. 49–60, 2014.

33. R. P. Duarte and C.-S. Bouganis, "Over-clocking of linear projection designs through device specific optimisations," in *21st Reconfigurable Architectures Workshop (RAW 2014)*, pp. 9–60, 2014.
34. R. P. Duarte and C.-S. Bouganis, "Pushing the performance boundary of linear projection designs through device specific optimisations (abstract only)," in *Proceedings of the 2014 ACM/SIGDA International Symposium on Field-programmable Gate Arrays*, FPGA '14, (New York, NY, USA), pp. 245–245, ACM, 2014.