ERROR ANALYSIS OF THE SQUARE ROOT OPERATION FOR THE PURPOSE OF PRECISION TUNING: A CASE STUDY ON K-MEANS

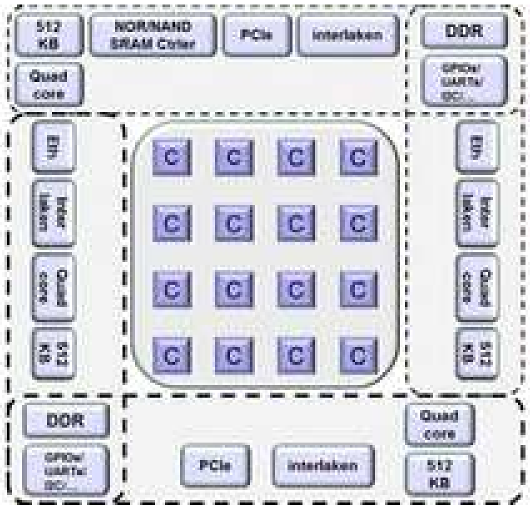
Oumaima Matoussi, Yves Durand, Olivier Sentieys, Anca Molnos

07/15/2019

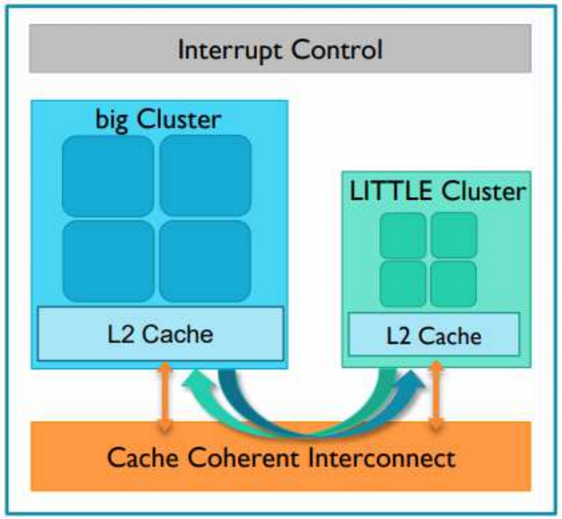
INTRODUCTION (1/2)

- **Volume and diversity of data have grown exponentially over the past few years**
- **Embedded systems try to keep pace with the constant growth of data**

➡ **Scale the technological parameters to increase performance while keeping energy consumption at bay.**



Kalray MPPA



ARM big.LITTLE

➡ **Many-core scaling is hitting a point of saturation**

INTRODUCTION (2/2)

- Many application domains (image/signal processing, machine learning, etc.) are tolerant to some degree of error
- Performance gains can be achieved at the application-level thanks to approximate computing
- Inexactness can be introduced in computations to reduce energy consumption

Compressed image
(32 colors)

Compressed image
(64 colors)

OUTLINE

- Introduction
- Approximate Computing
- Error Analysis of the Square Root Operation
- Application to K-means
- Experimental Results
- Conclusion

OUTLINE

- Introduction
- **Approximate Computing**
- Error Analysis of the Square Root Operation
- Application to K-means
- Experimental Results
- Conclusion

APPROXIMATE COMPUTING (1/2)

- **Approximate computing is an energy-efficient computing paradigm that exploits applications' tolerance to error**

Approximate Computing Approaches

HW approaches:
Approximate arithmetic units
(adders, multipliers, etc.)

SW approaches:
Approximate programming techniques
(precision tuning, loop perforation, etc.)

- **A way to introduce inexactness in an application is by precision reduction of FLP variables and computations**
- **A FLP number f is represented by an exponent e, a mantissa m and a sign bit s: f = $(-1)^{s}$ * m * 2^{e}**
- **Precision is the number of bits of the mantissa**

APPROXIMATE COMPUTING (2/2)

- **Precision tuning consists in finding the optimal bit width of FLP variables that:**
  - **Minimizes energy cost and**
  - **Maintains reasonable computational accuracy**

Precision Tuning

FLP Simulation techniques:
FLP with adjustable bit widths
e.g. MPFR library
Very large search space

Analytical techniques:
A mathematical formula
A one-time effort
Only works with smooth operations
(e.g. addition, multiplication, etc.)

- **Existing analytical work focuses on smooth operations, precisely addition and multiplication**

⇒ **Error analysis of square root operation is lacking**

⇒ **How to deal with applications containing both smooth and non-smooth operations?**

OUTLINE

- Introduction
- Approximate Computing
- **Error Analysis of the Square Root Operation**
- Application to K-means
- Experimental Results
- Conclusion

ERROR ANALYSIS OF THE SQUARE ROOT OPERATION (1/3)

- The square root operation $y = \sqrt{a}$, a>0 is implemented using the Newton Raphson iteration

$$y_{n+1} = \frac{1}{2}(y_n + \frac{a}{y_n})$$

- Two types of error are investigated:
  - **Algorithmic deviation**: caused by the Newton Raphson approach
  - **Round-off error**: caused by FLP representation

➡ **At which Newton Raphson iteration is it preferable to stop the computations for a specific precision p?**

ERROR ANALYSIS OF THE SQUARE ROOT OPERATION (2/3)

- **Bounding the round-off error:**

$$\boldsymbol{\delta \leq n \times 3\varepsilon + \varepsilon,\ \varepsilon \leq \varepsilon_m}$$

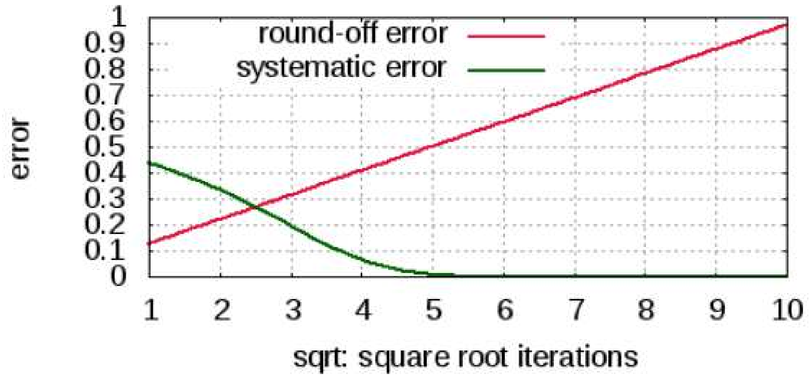
$\boldsymbol{\varepsilon_m}$: **machine epsilon**

n: $\boldsymbol{number\ of\ iterations}$

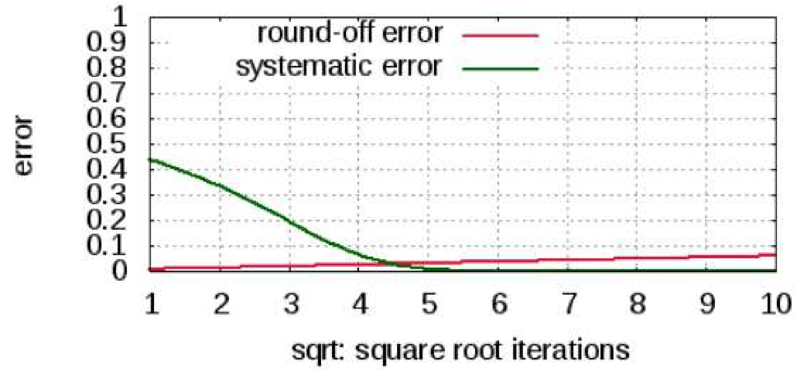
- **Bounding the systematic error: the relative systematic error in computing** $\mathbf{y} = \sqrt{\boldsymbol{a}}$ **at iteration i**: $\mathbf{0}..\boldsymbol{n}$

$$\boldsymbol{\frac{|y_i - \sqrt{a}|}{\sqrt{a}} \leq \frac{1}{2} \times \left(\frac{7}{8}\right)^{2^i - 1}}$$

ERROR ANALYSIS OF THE SQUARE ROOT OPERATION (3/3)



(a) p=4



(b) p=9

- **The round-off error increases proportionally to the number of sqrt iterations**
- **The systematic error declines as the number of iterations grows**

➡ The intersection between the 2 lines indicates the optimal number of iterations

OUTLINE

- Introduction
- Approximate Computing
- Error Analysis of the Square Root Operation
- **Application to K-means**
- Experimental Results
- Conclusion

APPLICATION TO K-MEANS (1/3)

- Apply our study of the square root in an application that contains both smooth and non-smooth operations

→ **Combine our analytical approach with a simulation approach**

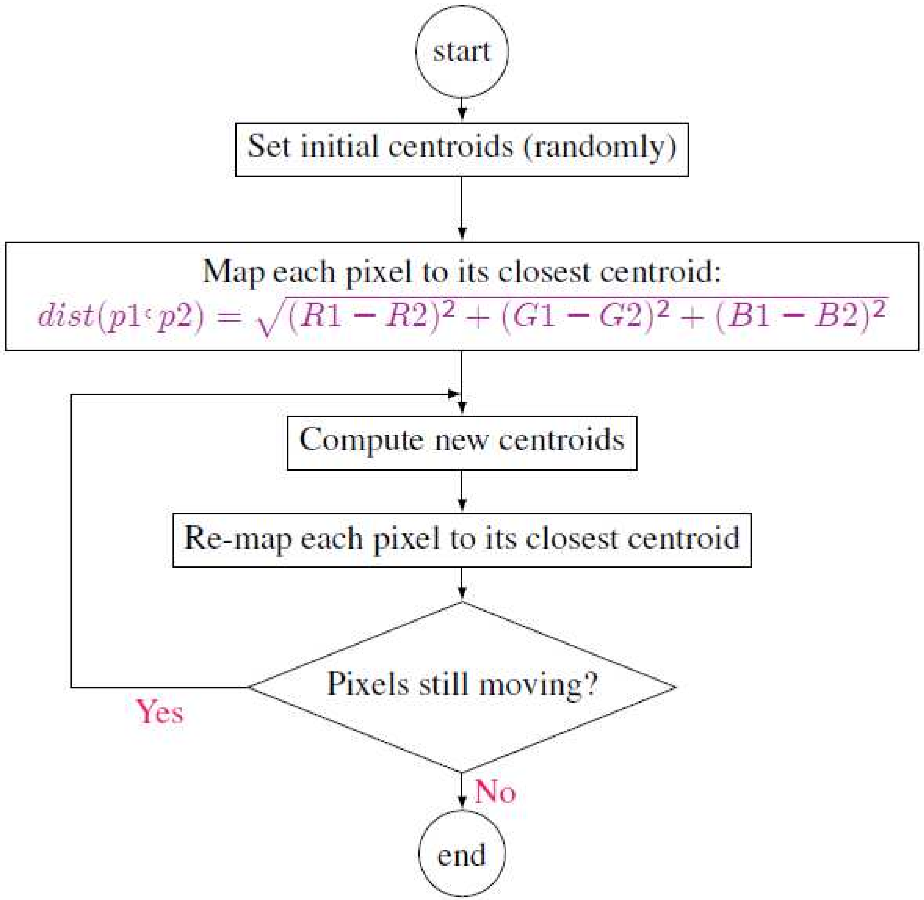
- Square root operations can be found in image/signal processing, spectrum analysis, clustering applications, etc.

  → **We chose K-means, a data clustering algorithm**

  → K-means is used to cluster a set of unlabeled data into k clusters based on data similarity

  → Similarity is determined using the Euclidean distance, which involves square root operations

APPLICATION TO K-MEANS (2/3)



Flowchart of K-means in the context of color quantization

APPLICATION TO K-MEANS (3/3)

- **We studied the sensitivity of K-means by arbitrarily varying the number of bits of the mantissa (2-23bits)**

- **We re-wrote K-means using the MPFR library**

```
mpfr_t var; // transform FLP variables into MPFR variables
mpfr_init2(var,4);   // assign precision 4 to var
mpfr_mul(var,var,var1,MPFR_RNDN); // transform operations into MPFR operations
```

- **We assign to each precision its corresponding number of square root iterations according to our analytical results**

- **We also varied K-means-specific parameters:**
  - K: number of clusters
  - n: number of iterations needed for the clustering process to converge

OUTLINE

- Introduction
- Approximate Computing
- Error Analysis of the Square Root Operation
- Application to K-means
- **Experimental Results**
- Conclusion

EXPERIMENTAL RESULTS (1/2)

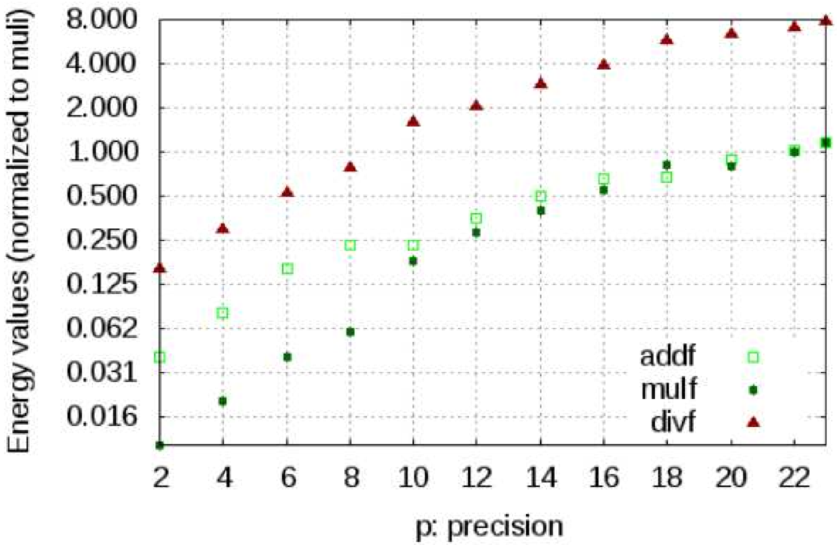
- **We transform the original source code of K-means by varying (k,n) for a given (p,sqrt), compile it to an ARM binary and check:**
  - Energy consumption (using measurements of ARM cortex-A7 and profile information)

$$E_{total} = \sum_{i=1}^{\#types} op_i \times e_i \ ,$$

op_i: nbr of operations of type i

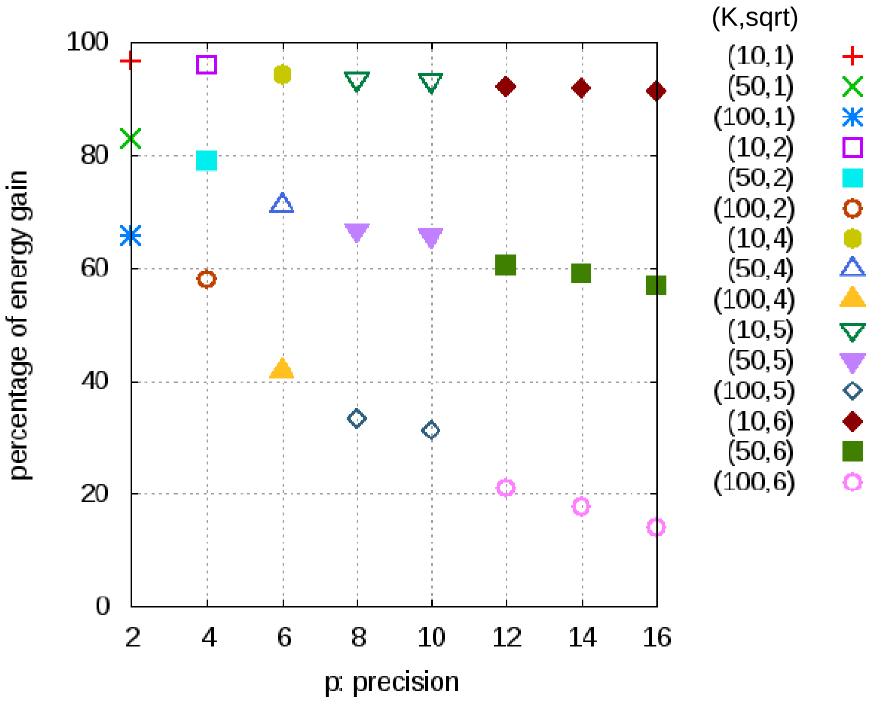
e_i: energy consumed per operation of type i

  - QoS using the SSIM index (perception-based metric)

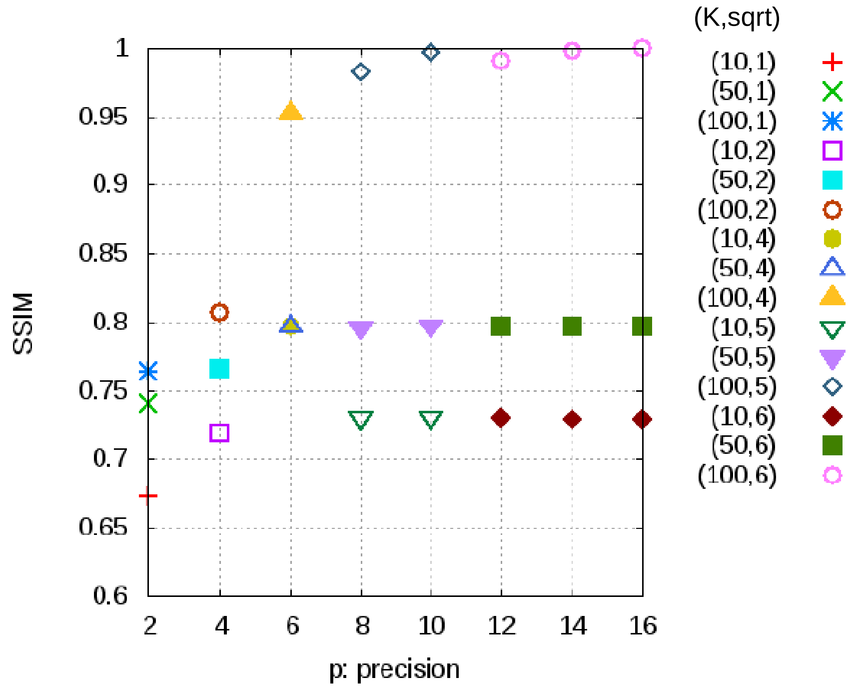


Normalized energy values for different precisions

EXPERIMENTAL RESULTS (2/2)



Percentage of energy gain



SSIM values

- **For an SSIM within [0.95,1], an energy gain of 41.87% is achieved with a (p=6,k=100,sqrt=4) configuration**

OUTLINE

- Introduction
- Approximate Computing
- Error Analysis of the Square Root Operation
- Application to K-means
- Experimental Results
- **Conclusion**

CONCLUSION

- An analytical error examination of the Newton Raphson approximation was proposed to optimize the sqrt implementation
- We associated to each precision its optimal number of Newton Raphson iterations
- We quantified the efficiency of the error bound in the context of K-means
- The approximated versions of K-means were compared to the exact version in terms of QoS and relative energy gain

Leti, technology research institute
Commissariat à l'énergie atomique et aux énergies alternatives
Minatec Campus | 17 rue des Martyrs | 38054 Grenoble Cedex | France
www.leti.fr