M03 A Comprehensive Analysis of Approximate Computing Techniques: From Component- to Application-Level

Session Type: Tutorial
Date: Monday, March 25, 2019
Time: 14:00-18:00
Location / Room: Room 5

Organisers
Daniel Menard, INSA Rennes/IETR, FR (Contact Daniel Menard [1])
Alberto Bosio, INL, FR (Contact Alberto Bosio [2])
Olivier Sentieys, INRIA, FR (Contact Olivier Sentieys [3])

A new design paradigm, Approximate Computing (AxC), has been established to investigate how computing systems can be more energy efficient, faster, and less complex. Intuitively, instead of performing exact computation and, consequently, requiring a high amount of resources, AxC aims to selectively relax the specifications, trading accuracy off for efficiency. It has been demonstrated in the literature the effectiveness of imprecise computation for both software and hardware components implementing inexact algorithms, showing an inherent resiliency to errors.

This tutorial introduces basic and advanced topics on AxC. We intend to follow a bottom-up approach: from component, up to application-level. More in detail, we will first present existing approximate computing techniques according to three levels: hardware, data and computation. At hardware level, functional approximation through inexact operators and voltage over-scaling will be detailed. At data level, approximation can be carried out by using efficient arithmetic, precision scaling, less data or less-up-to-date data. We will present some compile-time results in terms of energy-efficiency, area, performance versus accuracy of computations when using customized arithmetic (fixed-point, floating-point) and also, we will try to derive some conclusions by comparing the different paradigms. At computation level, algorithmic transformations are used to reduce complexity by skipping or approximating parts of the processing. The concepts of loop perforation, early termination, memoization, and computation approximation will be detailed.

The second part of the tutorial is dedicated to methods and tools to exploit efficiently approximate computing. First of all, the different approaches to analyze approximation effects on application quality will be described. Then the complex problem of word-length optimization for fixed-point and floating-point is considered. Finally, the different frameworks for design space exploration will be detailed.

The last part of the tutorial is devoted to present how AxC paradigm can be exploited in the context of safety-critical applications. More in particular, the tutorial will show how design efficient and low-cost fault tolerance mechanisms.

The outline of the presentation is as follows

- General introduction Motivations
  - Techniques for approximate computing
    - Data level approximation
    - Hardware level approximation
    - Computation level approximation
  - Methods and tools for approximate computing
    - Analysis of approximation effect on application quality
    - Word-length optimization for fixed-point and floating-point
    - Design space exploration
- Approximate Computing for Safety-Critical Applications
  - Analysis of approximation effect on application quality
  - Word-length optimization for fixed-point and floating-point
  - Design space exploration

Agenda

<table>
<thead>
<tr>
<th>Time</th>
<th>Label</th>
<th>Session</th>
</tr>
</thead>
<tbody>
<tr>
<td>13:30</td>
<td>M03.1</td>
<td>Tutorial and Conference Registration</td>
</tr>
<tr>
<td>14:00</td>
<td>M03.2</td>
<td>Tutorials start</td>
</tr>
<tr>
<td>15:30</td>
<td>M03.3</td>
<td>Coffee Break for Tutorials</td>
</tr>
<tr>
<td>18:00</td>
<td>M03.4</td>
<td>Tutorials end</td>
</tr>
<tr>
<td>18:00</td>
<td>M03.5</td>
<td>Welcome Reception &amp; PhD Forum</td>
</tr>
</tbody>
</table>

Source URL: https://past.date-conference.com/date19/conference/tutorial-m03

Links:
[1] https://past.date-conference.com/date19/user/22232/contact_form
[2] https://past.date-conference.com/date19/user/762/contact_form
[3] https://past.date-conference.com/date19/user/4747/contact_form