Optimizing Precision for High-Performance, Robust, and Energy-Efficient Computations
Roman Iakymchuk, Stef Graillat, Fabienne Jézéquel, Daichi Mukunoki, Toshiyuki Imamura, Yiyu Tan, Atsushi Koshiba, Jens Huthmann, Kentaro Sano, Norihisa Fujita, et al.

To cite this version:
Roman Iakymchuk, Stef Graillat, Fabienne Jézéquel, Daichi Mukunoki, Toshiyuki Imamura, et al.. Optimizing Precision for High-Performance, Robust, and Energy-Efficient Computations. International Conference on High Performance Computing in Asia-Pacific Region, Jan 2020, Fukuoka, Japan. hal-02401813

HAL Id: hal-02401813
https://hal.science/hal-02401813
Submitted on 15 May 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Optimizing Precision for High-Performance, Robust, and Energy-Efficient Computations

Introduction
In numerical computations, the precision of floating-point computations is a key factor to determine the performance (speed and energy-efficiency) as well as the reliability (accuracy and reproducibility). However, the precision generally plays a contrary role for both. Therefore, the ultimate concept for maximizing both at the same time is the optimized/reduced precision computation through precision-tuning, which adjusts the minimal precision for each operation and code. Several studies have been already conducted for it so far, but the scope of those studies is limited to the precision-tuning alone. Instead, we propose a more broad concept of the optimized/minimal precision and robust (numerically reliable) computing with precision-tuning, involving both hardware and software stack.

Our Proposal
Minimal-Precision Computing
Minimal-precision computing is both reliable (aka robust and survivable) and energy-efficient. It is particularly useful in low-power computing such as in the IoT as well as in high-performance computing such as exascale HPC because it can request the accurate result as well as being energy-efficient.

System Stack

<table>
<thead>
<tr>
<th>System Stack</th>
<th>FP64, FP32, FP16, FP8, FP4</th>
<th>MPPACK (for tuning)</th>
<th>CADNA, SAM</th>
<th>Precision Tuning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Numerical Validation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CADNA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FPGA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Heterogeneous System</td>
<td>Nymble</td>
<td>FPGA</td>
<td>Compiler &amp; Tools</td>
<td></td>
</tr>
<tr>
<td>FPGA</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hardware</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arithmetic Library</td>
<td>MPPACK</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FP64, FP32, FP16, FP8, FP4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

High-performance
Performance can be improved through the minimal-precision as well as fast numerical libraries and accelerators.

Energy-Efficient
Through the minimal-precision as well as the energy-efficient hardware architecture with FPGA and GPU.

Robust (Numerically Reliable)
To tune the minimal-precision, precision-tuning is processed based on numerical validation, guaranteeing also reproducibility.

General
Our scheme is applicable for any floating-point computations. It contributes to low development cost and sustainability (easy maintenance and system portability).

Comprehensive
We propose a total system from the precision-tuning to the execution of the tuned code, combining heterogeneous hardware and hierarchical software stack.

Realistic
Our system can be realized by combining available in-house technologies.

Available Components
(1) Precision-tuning with numerical validation based on stochastic arithmetic
- The uncertainty can be estimated stochastically with a reasonable cost for details, see [1] "Stochastic Arithmetic Tools" at the bottom of this.
- General scheme applicable for any floating-point computations.

Tools:
- PROMISE [17] (based on a stochastic arithmetic framework), CADNA [15], SAM [12], etc.
- Related works (not validation-based):
  - PathCov [20], GP_MAC [21], etc.

(2) Arbitrary-precision libraries and fast accurate numerical libraries
- Reduced/raised-precision with FP64/FP32/FP16 enables us to improve performance & energy-efficiency
- High-precision libraries and fast accurate computation methods have been developed for reliable & reproducible computation

Tools:
- High-precision arithmetic: binary128 (double precision, GPU) [1], [2]
- Basic Linear Algebra Package (BLAS) [3], [4], [5], [6], [7], etc.
- Accurate summation: ACUMSumD [8], DStrSum scheme [9], etc.
- Numerical libraries: QPBLAS [10], XBLAS [11], ExaBLAS [12], etc.

(3) Field-Programmable Gate Array (FPGA) with High-Level Synthesis (HLS)
- HLS enables us to implement any operations on hardware, including arbitrary precision operations
- HLS enables us to use OpenCL through existing programming languages such as C++ and OpenMP
- CADNA can be tuned to perform arbitrary-precision computations on hardware efficiently (high-performance and energy-efficient)

Tools:
- FPGA: SPGEn [14], Symple [15], etc.
- Custom floating-point operation generator: PoPulA [16], etc.

Our Contributions
Stochastic Arithmetic Tools
Decompose Stochastic Arithmetic ([24], [25], [26], [27]) into two components (i.e., the number of correct bits and the roundoff error), and only accuracy by executing the code 3 times on the basis of non-Round mode (NR-mode). This is a general scheme applicable for any floating-point computation on heterogeneous algorithms and no code modification (e.g., on heterogeneous systems) is a lightweight approach in terms of performance, while excellent development cost compared to the other numerical validation / simulation methods.

CADA & SAM (Sorbonne University)
- CADA (Control of Accuracy and Debugging for Numerical Applications) is a GPU library for FP64/FP32/FP16/FP8/FP4.
- CADNA can be used on CPUs in Mooract/C++ codes with [23].
- SAM (Stochastic Arithmetic in Multiprecision) is a CUDA library for arbitrary-precision with MADARA.

PFPGA as an Arbitrary-Precision Computing Platform
FPGA enables us to implement arbitrary-precision on hardware. High-level synthesis (HLS) enables us to program FPGA (in OpenCL), however, compiling arbitrary-precision code and obtaining high performance is still challenging. High-performance computing with FPGA & CPU/GPU is also a challenge.

QPBlAS (TU Darmstadt, RIKEN)
- QPBlAS is a library for arbitrary-precision floating point computations on CPUs.
- It directly accepts C codes and has already successfully used on TensorFlow.
- It is more suitable for non-linear memory access patterns like with real-world data structures.

Cygnus University of Tsukuba
- Cygnus is a research group in Denmark, creating a reference system called graVity (GraVity: High-Performance and Robust Numerical Libraries). [8]
- They provide a high-performance library called SPGen which is suitable for arbitrary-precision.
- Each Strato 3FPGA has four coherent interlinks at 30Gbps (4) FPGA-rocket I/O network for communication.
- Cygnus system targets such heterogeneous systems with FPGA.

Fast and Accurate Numerical Libraries
Arbitrary-precision arithmetic is performed using MPIR or GMP, but the performance is very low. At present, we are working on efficient multiprecision libraries which are fully applicable to high-performance arithmetic or algorithms. Some software also support GPU acceleration.

QPBlAS (TU Darmstadt, RIKEN)
- QPBlAS is an accurate & reproducible BLAS based on floating-point expression (FP) library. It is based on modular and efficient BLAS library ([28]), which supports CUDA. QPBlAS has a high performance and accuracy on various platforms.

OzBlAS (TUC, TU Darmstadt, RIKEN)
- OzBlAS is an accurate & reproducible BLAS using Cuckoo scheme ([19]) which is an accurate memory-multiplication method based on the non-invariant transformation of dot-product calculation. OzBlAS is based on the following concepts:
- The required accuracy must be taken into account.
- If possible, it considers to utilize FPGA (as heterogeneous computing).

Conclusion & Future Work
We proposed a new systematic approach for minimal-precision computations. This approach is robust, general, comprehensive, high-performance, and realistic. Although the proposed system is still in development, it can be constructed by combining already available (developed) in-house technologies and extending them. Our ongoing step is to demonstrate the system on a proxy application.

References

Acknowledgements:
This work was partially supported by the France-Japan research project, the Numerical Analysis Working Group, and the Japan Society for the Promotion of Science (JSPS) KAKENHI Grant (26280033). The authors would like to thank the University of Tsukuba, Kyoto University, and RIKEN for their support.

HPC Asia 2020 – International Conference on High-Performance Computing in Asia-Pacific Region, Fukuoka, Japan, January 15-17, 2020