


Using precision emulation within mixed-precision algorithms

Introduction

Efficient low-precision arithmetic units are becoming increasingly widespread on both high-performance or even commodity computers. These include half-precision floating-point units that employ 16 bits but even lower-precision formats using 8 or 4 bits are becoming more and more common; in the same way, units with low-precision (8 bit) integer arithmetic are also gaining popularity. This trend, especially observable on modern GPUs, is motivated not only by some technological issues (for example, the end of Moore's law) but also by the recent outbreak of artificial intelligence and machine learning that commonly rely on low-precision computations. These arithmetic units offer extremely high performance (up to 100 times higher than double precision), lower memory and energy consumption, and are therefore very attractive for scientific computing and AI developers and users. Although these units can be employed in AI or ML workflows in a relatively straightforward way, their use in scientific computing applications (e.g., numerical simulation) poses numerous issues because they do not deliver sufficient accuracy. Therefore a considerable research effort is being deployed by both the academic and industrial community to figure out ways to take advantage of the performance offered by these units in scientific computing algorithms without compromising their accuracy and robustness. Mixed-precision algorithms [1] are one result of these efforts: they use different arithmetics (with different precision) in different steps in order to improve performance while preserving satisfactory (with respect to the application needs) accuracy and robustness. Mixed-precision iterative refinement [2,3] is one popular example of mixed precision algorithms for the solution of systems of linear equations where the bulk of computations is done in low precision to compute an approximate solution which is then refined in successive iterations that employ cheap high-precision computations. Another example is the adaptive-precision pivoted QR factorization for computing low-rank approximations of large datasets [4] where the precision of computation is progressively reduced as the factorization proceeds depending on the data spectral properties.

Other types of algorithms, instead, fall into a different category that is commonly referred to as "precision emulation". In these algorithms, low-precision (floating-point or even integer) units are used to emulate computations in higher precision. For example, multiword arithmetic can be used to compute the multiplication of two matrices using 8-bit integer units while providing double-precision accuracy [5]. In these methods, the matrices are approximately decomposed (or "sliced") into multiple terms that can be multiplied without

rounding error; moreover, the decomposition error can be reliably controlled by the number of slices. Even if these algorithms incur a higher operational complexity, the overall execution time can be reduced due to the much higher performance on low-precision (integer) units.

Goals of the internship

This internship focuses on exploring the combined use of precision-emulation methods within mixed-precision algorithms. Indeed, one attractive feature of precision-emulation methods is that they can virtually emulate a continuous level of arbitrary accuracy, including floating-point arithmetics for which no hardware support is available. Therefore, their use within mixed-precision algorithms gives rise to an extremely large number of precision combinations, each with its own properties in terms of performance and accuracy or robustness. More precisely, the two mixed-precision algorithms mentioned above will be targeted, namely, mixed-precision iterative refinement and adaptive-precision pivoted QR factorization.

For all these algorithms, the internship will involve, on the one hand, carrying out rigorous error analyses to provide guarantees on the quality of the final solution, and on the other hand, developing carefully optimized codes to make the most of their performance potential.

Context / work environment

This internship will be carried out in the context of the French NumPEx research progaellem (https://numpex.org/) on exascale high performance computing; furthermore it will be carried out in collaboration with the NVIDIA company.

- Location: IRIT laboratory in Toulouse or LIP6 laboratory in Paris or LIP laboratory in Lyon
- Salary: ~600/month
- Period: start date around March or April 2026; duration 5 or 6 months
- Contact: Alfredo Buttari (alfredo.buttari@irit.fr) or Theo Mary (theo.mary@lip6.fr)

References

[1] Nicholas J. Higham and Theo Mary. Mixed precision algorithms in numerical linear algebra, Acta Numerica, 31:347–414 (2022). https://hal.archives-ouvertes.fr/hal-03537373

[2] Azzam Haidar, Harun Bayraktar, Stanimire Tomov, Jack Dongarra and Nicholas J. Higham. Mixed-precision iterative refinement using tensor cores on GPUs to accelerate solution of linear systems, Proc. R. Soc. A. 476. https://doi.org/10.1098/rspa.2020.0110

[3] Patrick R. Amestoy, Alfredo Buttari, Nicholas J. Higham, Jean-Yves L'Excellent, Theo Mary and Bastien Vieublé. Five-precision GMRES-based Iterative Refinement, SIAM J. Matrix Anal. Appl., 45(1), 529–552 (2024). https://hal.archives-ouvertes.fr/hal-03190686

[4] Alfredo Buttari, Theo Mary & André Pacteau. Truncated QR factorization with pivoting in mixed precision, SIAM J. Sci. Comput., 47(2), B382–B401 (2025). https://hal.science/hal-04490215

[5] Hiroyuki Ootomo, Katsuhisa Ozaki, and Rio Yokota. DGEMM on integer matrix multiplication unit. The International Journal of High Performance Computing Applications.38(4), 297-313 (2024). https://doi.org/10.1177/10943420241239588