Trade-off between accuracy and time for automatically generated summation algorithms

Philippe Langlois, Matthieu Martel and Laurent Thévenoux

DALI Research Team
Université de Perpignan Via Domitia
52 Avenue Paul Alduy
66860 Perpignan Cedex
France
e-mail:{langlois, matthieu.martel, laurent.thevenoux}@univ-perp.fr

Abstract

We focus on numerical algorithms for which performances and accuracy do not cohabit well in practice [2]. A widely studied example is the floating-point summation – see for instance [3, 6, 7, 5].

In order to increase parallelism, expressions are reparsed implicitly using arithmetic properties like associativity or distributivity. In IEEE-754 floating-point arithmetic [1], these laws do not hold any longer for any arithmetic expression. The numerical accuracy of the algorithm may be strongly sensitive to reparsing. So increasing the parallelism of some algorithm may decrease its numerical accuracy, and, conversely, improving the accuracy of some computation may reduce its parallelism. We present an experimental analysis of this problem for the floating-point summation.

Our approach consists in performing an exhaustive study. First we generate all the algorithms mathematically equivalent to the original one and compatible with some relaxed time constraints. Then we compute the worst errors which may arise during their evaluation for several relevant sets of data. Let n be the number of additions and let k be an arbitrary constant. We propose to restrict the search of accurate computation within the three enclosed sets of algorithms having a computing tree of height smaller or equal to, respectively, $\lfloor \log(n) \rfloor + 1$, $\lfloor \log(n) \rfloor + k$ and $k \times \lfloor \log(n) \rfloor$.

We observe that in most cases, the highest level of parallelism –$O(\log(n))$ computing levels– does not allow us to compute the most accurate results. Nevertheless a less high but still reasonable level of parallelism, like levels $O(\log(n) + k)$ or $O(k \times \log(n))$, produces accurate results. Moreover the proportion of optimal algorithms which respect to the best accuracy is tiny.

Our main conclusion is that relaxing very slightly the time constraints by choosing algorithms whose critical paths are a bit longer than for optimal one makes it possible to strongly optimize the accuracy. This matter of fact is illustrated using various data-sets, most of them being ill-conditioned. We observe that the higher the level of parallelism, the

harder it is to find an accurate summation algorithm. But if we relax the time constraint, namely the parallelism, it is easier to get an optimally accurate algorithm.

We extend these results to the case of bounded parallelism and to compensated algorithms. For bounded parallelism we show that more accurate algorithms whose critical path is not optimal can be executed in as many cycles as optimal algorithms, *e.g.* on VLIW architectures like the ones uses in [4]. Concerning compensation, we show that classical accurate summation algorithms can be discovered automatically by inserting systematically compensations and then reparsing the resulting expression.

References

[1] ANSI/IEEE. *IEEE Standard for Binary Floating-point Arithmetic*, revision of std 754-1985 edition, 2008.

[2] J. Demmel. Trading off parallelism and numerical stability, 1992. http://www.eecs.berkeley.edu/Pubs/TechRpts/1992/CSD-92-702.pdf.

[3] N. J. Higham. *Accuracy and Stability of Numerical Algorithms.* Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, second edition, 2002.

[4] C.-P. Jeannerod and G. Revy. Optimizing correctly-rounded reciprocal square roots for embedded VLIW cores. In *Proceedings of the 43rd Asilomar Conference on Signals, Systems, and Computers (Asilomar'09)*, Pacific Grove, CA, USA, november 2009.

[5] P. Langlois, M. Martel, and L. Thevenoux. Accuracy Versus Time: A Case Study with Summation Algorithms. http://hal.archives-ouvertes.fr/hal-00477511/PDF/main.pdf, 2010.

[6] T. Ogita, S. M. Rump, and S. Oishi. Accurate sum and dot product. *SIAM J. Sci. Comput.*, 26(6):1955–1988, 2005.

[7] S. M. Rump. Ultimately fast accurate summation. *SIAM J. Sci. Comput*, 31(5):3466–3502, 2009.

Keywords: accuracy, summation, parallelism, IEEE-754 floating-point arithmetic.