Numerical Software: Design, Analysis and Verification
4–6 July 2012, Santander (Spain).

Less Hazard and More Scientific Research for the Computing Time of Summation Algorithms

Philippe Langlois[1], Bernard Goossens[1], David Parello[1] and Kathy Porada[2]

[1] *Univ. Perpignan Via Domitia, Digits, Architectures et Logiciels Informatiques, F-66860, Perpignan , France. Univ. Montpellier II, Laboratoire d'Informatique Robotique et de Microélectronique de Montpellier, UMR 5506, F-34095, Montpellier, France. CNRS, Laboratoire d'Informatique Robotique et de Microélectronique de Montpellier, UMR 5506, F-34095, Montpellier, France,*

[2] *Univ. Perpignan Via Domitia, Digits, Architectures et Logiciels Informatiques, F-66860, Perpignan, France,*

emails: `langlois@univ-perp.fr`, `goossens@univ-perp.fr`, `david.parello@univ-perp.fr`, `katarzyna.porada@etudiant.univ-perp.fr`

Abstract

Several accurate algorithms to sum IEEE-754 floating point numbers have been recently published. The contributions by Rump, Ogita and Oishi [3, 4, 2] and the newest ones proposed by Zhu and Hayes [5, 6] are examples of accurate summation algorithms. Some of these actually compute the correct rounding of the exact sum, *i.e.*, the most accurate value with respect to the finite precision of the floating point arithmetic. This computed sum does not suffer anymore from the condition number of the summation. In such cases, the run-time performances and the memory print become the discriminant properties to decide which algorithm to choose.

In this talk we focus the problem of presenting reliable measures of the run-time performances of such core algorithms. As Rump writes in [2], "Measuring the computing time of summation algorithms in a high-level language on today's architectures is more of a hazard than scientific research." We introduced PerPI, a software tool that automatizes an almost machine independent analysis based on the instruction-level parallelism of the algorithm [1]. We analyze the previously mentioned accurate summation algorithms. We discuss whether PerPI provides a more reliable performance analysis and how to improve the confidence level of future contributions in this area.

Key words: floating-point arithmetic, accurate summation, performance evaluation, reproducibility, PerPI.

References

[1] B. Goossens, Ph. Langlois, D. Parello, and E. Petit. PerPI: A Tool to Measure Instruction Level Parallelism. *Applied Parallel and Scientific Computing.* LNCS-7133, pp. 270–281. Springer, 2012.

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

[3] S.M. Rump, T. Ogita, and S. Oishi. Accurate floating-point summation – part I: Faithful rounding. *SIAM J. Sci. Comput.*, 31(1):189–224, 2008.

[4] S.M. Rump, T. Ogita, and S. Oishi. Fast high precision summation. *Non Linear Theory and its Application*, pp. 1–23. IEICE, 2010.

[5] Y.-K. Zhu and W. B. Hayes. Correct rounding and hybrid approach to exact floating-point summation. *SIAM J. Sci. Comput.*, 31(4):2981–3001, 2009.

[6] Y.-K. Zhu and W. B. Hayes. Algorithm 908: Online exact summation of floating-point streams. *ACM Transactions on Mathematical Software*, 37(3):37:1–37:13, Sept. 2010.