Pergamon

Appl. Math. Lett. Vol. 11, No. 6, pp. 49–50, 1998
© 1998 Elsevier Science Ltd. All rights reserved
Printed in Great Britain
0893-9659/98 $19.00 + 0.00

PII: S0893-9659(98)00101-3

Single Precision Computation of the Sign of Algebraic Predicates

V. Y. PAN
Mathematics and Computer Science Department
Lehman College, City University of New York, Bronx, NY 10468, U.S.A.
vpan@lcvax.lehman.cuny.edu

(Received and accepted October 1997)

Abstract—We simplify and improve our techniques of the association of long integers with polynomials for computations in the ring of integers and apply these techniques to the computation of the signs of matrix determinants, Sturm sequences, and other algebraic and geometric predicates. © 1998 Elsevier Science Ltd. All rights reserved.

Keywords—Integers, Associated polynomials, Single precision computations, Signs of matrix determinants, Sign of algebraic and geometric predicates.

1. INTRODUCTION

Our subject is the computation of a sign of a fixed expression in the ring of integers. This is a central problem of geometric and algebraic computations, which includes such major special cases as the computation of the *Cauchy index* of a rational fraction (covering in particular *Sturm sequences*) and of *the sign of a matrix determinant.* We refer the reader to [1,2], and references therein on background and recent progress.

The major practical requirement to the solution algorithms is to ensure the correct output by computing with a fixed single or double precision, since computer implementations of variable and/or multiple precision, as well as infinite precision symbolic computation are currently much more costly in terms of computer time and memory involved.

In this paper, we extend the technique of the *association of long integers with polynomials* proposed in [2, Section 11], but employ it more directly towards the goal stated above, to simplify the approach of [2] and to accentuate its power. Whereas this technique was developed and elaborated in [2] as a part of Chinese remainder algorithm, our present simplified version can be used in any algorithm performing in the ring of integers. This version is close to the usual techniques of multiprecision computation (see [3]), but we confine the entire computation to operations in the ring of univariate polynomials where all coefficients are single precision integers. Unlike much more involved and expensive packages of subroutines available for symbolic computation, such a simplified version can be made accessible via a few simple microcodes. Here is a formal description of this version.

DEFINITION. *(See [2].) For a fixed integer base $b > 1$, for any integer a, and any polynomial $a(x)$ with integer coefficients such that $a(b) = a$, define the unique b-associate $r_{b,a}(x) = \operatorname{sign}(a) \sum_i r_i x^i$ of a and $a(x)$ (also called their b-reduction) satisfying $|r_{b,a}(b)| = |a|$, $0 \le r_i < b$ for all i.*

Supported by NSF Grant CCR 9625344 and PSC CUNY Award 668385.

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TeX

By definition, $r_{b,a}(x) = 0$ iff $a = 0$, and otherwise, $\mathrm{sign}(a)$ coincides with the common sign of all nonzero coefficients of $r_{b,a}(x)$.

Suppose that we have an expression $f(a_1, \dots, a_n)$ in the ring $\mathbf{Z}$ of integers and an algorithm that reduces the computation of $f(a_1, \dots, a_n)$ to a sequence of ring operations (that is, +, −, and *) and that we seek $\mathrm{sign}(f(a_1, \dots, a_n))$. Then we may fix a base b, replace the input integers $a_1, \dots, a_n$ by their b-associates, perform the ring operations of the algorithm with the b-associates of all the integers involved, and read off $\mathrm{sign}(f(a_1, \dots, a_n))$ from the sign of the coefficients of the output b-associate. Similarly, we proceed where we seek the signs of several expressions defined in the ring of integers.

To perform the ring operations $(+, -, *)$ with the b-associates of integers, one may employ the known algorithms for integers [3, pp. 251–255, 278–279].

The choice of the base $b = 2^{p/(2d+2)}$, where d is the maximum degree of the polynomials involved as the operands of the computations, ensures that such computations with a fixed or double binary precision p, but with no rounding always produce correct output. The maximum degree d is estimated prior to the computation. $d = \lceil \log_b M \rceil$, where M is the maximum absolute value of all integers involved in the original algorithm (over integers). In the particularly important case of computing the sign of $\det A$, the determinant of a matrix, the value M can be estimated based on Hadamard's bound. For matrices A of small sizes, one may compute $\det A$ from its recursive decomposition into linear combinations of the minors (subdeterminants). For larger sizes, Bareiss's and other fraction-free methods supply M (via Hadamard's bound) as a by-product of the factorization of A, which gives us $\det A$.

REFERENCES

1. H. Brönnimann, I.Z. Emiris, V. Pan and S. Pion, Computing exact geometric predicates using modular arithmetic with single precision, In *Proc. 13th Annual ACM Symposium on Computational Geometry*, pp. 174–182, ACM Press, New York, (1997).
2. V.Y. Pan, Y. Yu and C. Stewart, Algebraic and numerical techniques for the computation of matrix determinants, *Computers Math. Applic.* **34** (1), 43–70 (1997).
3. D.E. Knuth, *The Art of Computer Programming: Seminumerical Algorithms*, 2, Addison-Wesley, Reading, MA, (1981).