CHECKSUM SCHEMES FOR FAULT TOLERANT SYSTOLIC COMPUTING

RICHARD P. BRENT, F. T. LUK, AND C. J. ANFINSON

Abstract

The weighted checksum scheme has been proposed as a low-cost fault tolerant procedure for parallel matrix computations. To guarantee multiple error detection and correction, the chosen weight vectors must satisfy some very specific properties regarding linear independence. We provide a theoretical framework for these properties, and prove that the exact number of errors can be determined for a distance $d + 1$ scheme if there is a maximum of $\lfloor d/2 \rfloor$ errors. We also derive a procedure for correcting the errors. Previous weight generating methods that fulfil the independence criteria have troubles with numerical overflow. We present a new scheme that generates weight vectors to meet the requirements regarding independence and to avoid difficulties with overflow.

Comments

Only the Abstract is given here. The full paper appeared as [1]. An earlier version appeared as [2]. For related work, see [3].

References

1. R. P. Brent, F. T. Luk and C. J. Anfinson, "Checksum schemes for fault tolerant systolic computing", *Mathematics in Signal Processing II* (edited by J. G. McWhirter), Clarendon Press, Oxford, 1990, 791–804. ISBN 0-19-853641-0. A modified version of "Weight selection for the checksum scheme", presented at the Conference on Mathematics in Signal Processing, Univ. of Warwick, UK, Dec. 1988. rpb112.
2. R. P. Brent, F. T. Luk and C. J. Anfinson, "Choosing small weights for multiple error detection", *Proceedings SPIE, Volume 1058, High Speed Computing II*, SPIE, Los Angeles, 1989, 130–136. rpb114.
3. D. L. Boley, R. P. Brent, G. H. Golub and F. T. Luk, "Error correction via the Lanczos process", *SIAM J. on Matrix Analysis* (Gene H. Golub birthday issue) 13 (1992), 312–332. rpb124.

(Brent) Australian National University, Canberra

(Luk, Anfinson) Cornell University, New York

1991 *Mathematics Subject Classification.* Primary 94B05; Secondary 68Q35, 94B35, 94B99.

Key words and phrases. Error correction, error detection, fault tolerance, algorithmic fault tolerance, Reed-Solomon code, BCH code, overflow.