COMPLEXITY MEASURES FOR MATRIX MULTIPLICATION ALGORITHMS

F. ROMANI (I)

ABSTRACT - A new class of algorithms for the computation of bilinear forms has been recently introduced [1, 5]. These algorithms approximate the result with an arbitrarily small error. Such approximate algorithms may have a multiplicative complexity smaller than exact ones. On the other hand any comparison between approximate and exact algorithms has to take into account the complexity-stability relations.

In this paper some complexity measures for matrix multiplication algorithms are discussed and applied to the evaluation of exact and approximate algorithms. Multiplicative complexity is shown to remain a valid comparison test and the cost of approximation appears to be only a logarithmic factor.

1. Introduction and preliminaries.

Consider the problem of computing the bilinear forms

\[ x^T A_h y \quad (h = 1, \ldots, p) \]

\[ x \text{ } n\text{-vector}, \text{ } y \text{ } m\text{-vector}, \text{ } A_h = \{a_{ij}^{(h)}\} \text{ } n \times m \text{ matrices.} \]

In [1, 5] those algorithms yielding an exact solution to this problem were called \( EC \)-algorithms (Exactly Computing). A bilinear \( EC \) algorithm is identified by three matrices

\[ U = \{u_{ir}\}, \ V = \{v_{ir}\}, \ W = \{w_{hr}\} \]

\[ (i = 1, \ldots, n; \ j = 1, \ldots, m; \ h = 1, \ldots, p; \ r = 1, \ldots, t) \]

satisfying the condition

---

Received April 10, 1979.
(I) Istituto di Elaborazione della Informazione, CNR - Pisa.
The bilinear forms are computed by the formula

\[
\sum_{r=1}^{t} u_{ir} v_{jr} w_{hr} = a_{ij}^{(h)}.
\]

and \( t \) is the number of non scalar multiplications.

APA-algorithms (Arbitrary Precision Approximating) were introduced to take advantage of the circumstance that the complexity may be reduced by allowing the result to be affected by an arbitrarily small error. A bilinear APA-algorithm is identified by three matrices \( U(\varepsilon), V(\varepsilon), W(\varepsilon) \) satisfying the condition

\[
\sum_{r=1}^{t} u_{ir} (\varepsilon) v_{jr} (\varepsilon) w_{hr} (\varepsilon) = a_{ij}^{(h)} + e_{ij}^{(h)} (\varepsilon) \quad (h = 1, \ldots, p)
\]

where \( E_{h}(\varepsilon) = \{ e_{ij}^{(h)} (\varepsilon) \} \) are matrices of continuous functions of \( \varepsilon \) and \( E_{h}(0) \) are null matrices. If the entries of \( U, V, W \) are powers of \( \varepsilon \) the \( E_{h}(\varepsilon) \) are polynomials in \( \varepsilon \).

**Remark:** In [1] an APA-algorithm for \( n \times n \) matrix multiplication with multiplicative complexity \( O(n^{2.779}) \) is presented.

In [5] it is shown that the existence of APA-algorithms requiring \( t' \) multiplications implies the existence of EC-algorithms \( t'(1+d) \) multiplications; \( d \) is the degree of the polynomial corrections \( E_{h}(\varepsilon) \).

Matrix multiplication (in the following \( MM \)) is a special case of the problem of computing bilinear forms. Moreover the three-way array \( A = \{ a_{ij}^{(h)} \} \) associated to a \( k^{d} \times k^{d} \) \( MM \) problem is the \( q \)-th tensorial power of the three-way array associated to the \( k \times k \) problem [5]. This fact supports the well known technique to derive a general \( n \times n \) algorithm from a \( k \times k \) algorithm by recursive partitioning [6,8]. As a matter of fact the recursive application of the same algorithm is equivalent to use a bilinear algorithm identified by matrices \( U(q), V(q), W(q) \) which are the \( q \)-th tensorial powers of \( U, V, W \).

In section 2 some complexity measures to evaluate general \( MM \) algorithms by taking into account numerical stability are discussed. This allows to compare the efficiency of EC-algorithms and APA-algorithms. An alternative technique to perform \( MM \) using APA-algorithms is also considered. Logarithms are to base 2 throughout this paper unless otherwise indicated.