Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

179

Victor Pan

How to Multiply Matrices Faster

Springer-Verlag
Berlin Heidelberg New York Tokyo

Lecture Notes in Computer Science

Edited by G. Goos and J. Hartmanis

179

Victor Pan

How to Multiply Matrices Faster

Springer-Verlag
Berlin Heidelberg New York Tokyo 1984

Author

Victor Pan
State University of
Department of Cor
1400 Washington A

CR Subject Classification (1982): F.2.1, G.1.3

ISBN 3-540-13866-8 Springer-Verlag Berlin Heidelberg New York Tokyo
ISBN 0-387-13866-8 Springer-Verlag New York Heidelberg Berlin Tokyo

Printing and binding: Beltz Offsetdruck, Hemsbach / Bergstr.
2146 / 3140-543210

Lecture Notes in Computer Science

Vol. 88: Mathematical Foundations of Computer Science 1980. Proceedings, 1980. Edited by P. Dembiński. VIII, 723 pages. 1980.

Vol. 89: Computer Aided Design - Modelling, Systems Engineering, CAD-Systems. Proceedings, 1980. Edited by J. Encarnacao. XIV, 461 pages. 1980.

Vol. 90: D. M. Sandford, Using Sophisticated Models in Resolution Theorem Proving.
XI, 239 pages. 1980

Vol. 91: D. Wood, Grammar and L Forms: An Introduction. IX, 314 pages. 1980.

Vol. 92: R. Milner, A Calculus of Communication Systems. VI, 171 pages. 1980.

Vol. 93: A. Nijholt, Context-Free Grammars: Covers, Normal Forms, and Parsing. VII, 253 pages. 1980.

Vol. 94: Semantics-Directed Compiler Generation. Proceedings, 1980. Edited by N. D. Jones. V, 489 pages. 1980.

Vol. 95: Ch. D. Marlin, Coroutines. XII, 246 pages. 1980.

Vol. 96: J. L. Peterson, Computer Programs for Spelling Correction VI, 213 pages. 1980.

Vol. 97: S. Osaki and T. Nishio, Reliability Evaluation of Some Fault-Tolerant Computer Architectures. VI, 129 pages. 1980.

Vol. 98: Towards a Formal Description of Ada. Edited by D. Bjørner and O. N. Oest. XIV, 630 pages. 1980.

Vol. 99: I. Guessarian, Algebraic Semantics. XI, 158 pages. 1981.

Vol. 100: Graphtheoretic Concepts in Computer Science. Edited by H. Noltemeier. X, 403 pages. 1981.

Vol. 101: A. Thayse, Boolean Calculus of Differences. VII, 144 pages. 1981.

Vol. 102: J. H. Davenport, On the Integration of Algebraic Functions. 1–197 pages. 1981.

Vol. 103: H. Ledgard, A. Singer, J. Whiteside, Directions in Human Factors of Interactive Systems. VI, 190 pages. 1981.

Vol. 104: Theoretical Computer Science. Ed. by P. Deussen. VII, 261 pages. 1981.

Vol. 105: B. W. Lampson, M. Paul, H. J. Siegert, Distributed Systems – Architecture and Implementation. XIII, 510 pages. 1981.

Vol. 106: The Programming Language Ada. Reference Manual. X, 243 pages. 1981.

Vol. 107: International Colloquium on Formalization of Programming Concepts. Proceedings. Edited by J. Diaz and I. Ramos. VII, 478 pages. 1981.

Vol. 108: Graph Theory and Algorithms. Edited by N. Saito and T. Nishizeki. VI, 216 pages. 1981.

Vol. 109: Digital Image Processing Systems. Edited by L. Bolc and Zenon Kulpa. V, 353 pages. 1981.

Vol. 110: W. Dehning, H. Essig, S. Maass, The Adaptation of Virtual Man-Computer Interfaces to User Requirements in Dialogs. X, 142 pages. 1981.

Vol. 111: CONPAR 81. Edited by W. Händler. XI, 508 pages. 1981.

Vol. 112: CAAP '81. Proceedings. Edited by G. Astesiano and C. Böhm. VI, 364 pages. 1981.

Vol. 113: E.-E. Doberkat, Stochastic Automata: Stability, Nondeterminism, and Prediction. IX, 135 pages. 1981.

Vol. 114: B. Liskov, CLU, Reference Manual. VIII, 190 pages. 1981.

Vol. 115: Automata, Languages and Programming. Edited by S. Even and O. Kariv. VIII, 552 pages. 1981.

Vol. 116: M. A. Casanova, The Concurrency Control Problem for Database Systems. VII, 175 pages. 1981.

Vol. 117: Fundamentals of Computation Theory. Proceedings, 1981. Edited by F. Gécseg. XI, 471 pages. 1981.

Vol. 118: Mathematical Foundations of Computer Science 1981. Proceedings, 1981. Edited by J. Gruska and M. Chytil. XI, 589 pages. 1981.

Vol. 119: G. Hirst, Anaphora in Natural Language Understanding: A Survey. XIII, 128 pages. 1981.

Vol. 120: L. B. Rall, Automatic Differentiation: Techniques and Applications. VIII, 165 pages. 1981.

Vol. 121: Z. Zlatev, J. Wasniewski, and K. Schaumburg, Y12M Solution of Large and Sparse Systems of Linear Algebraic Equations. IX, 128 pages. 1981.

Vol. 122: Algorithms in Modern Mathematics and Computer Science. Proceedings, 1979. Edited by A. P. Ershov and D. E. Knuth. XI, 487 pages. 1981.

Vol. 123: Trends in Information Processing Systems. Proceedings, 1981. Edited by A. J. W. Duijvestijn and P. C. Lockemann. XI, 349 pages. 1981.

Vol. 124: W. Polak, Compiler Specification and Verification. XIII, 269 pages. 1981.

Vol. 125: Logic of Programs. Proceedings, 1979. Edited by E. Engeler. V, 245 pages. 1981.

Vol. 126: Microcomputer System Design. Proceedings, 1981. Edited by M. J. Flynn, N. R. Harris, and D. P. McCarthy. VII, 397 pages. 1982.

Voll. 127: Y.Wallach, Alternating Sequential/Parallel Processing. X, 329 pages. 1982.

Vol. 128: P. Branquart, G. Louis, P. Wodon, An Analytical Description of CHILL, the CCITT High Level Language. VI, 277 pages. 1982.

Vol. 129: B. T. Hailpern, Verifying Concurrent Processes Using Temporal Logic. VIII, 208 pages. 1982.

Vol. 130: R. Goldblatt, Axiomatising the Logic of Computer Programming. XI, 304 pages. 1982.

Vol. 131: Logics of Programs. Proceedings, 1981. Edited by D. Kozen. VI, 429 pages. 1982.

Vol. 132: Data Base Design Techniques I: Requirements and Logical Structures. Proceedings, 1978. Edited by S.B. Yao, S.B. Navathe, J.L. Weldon, and T.L. Kunii. V, 227 pages. 1982.

Vol. 133: Data Base Design Techniques II: Proceedings, 1979. Edited by S.B. Yao and T.L. Kunii. V, 229–399 pages. 1982.

Vol. 134: Program Specification. Proceedings, 1981. Edited by J. Staunstrup. IV, 426 pages. 1982.

Vol. 135: R.L. Constable, S.D. Johnson, and C.D. Eichenlaub, An Introduction to the PL/CV2 Programming Logic. X, 292 pages. 1982.

Vol. 136: Ch. M. Hoffmann, Group-Theoretic Algorithms and Graph Isomorphism. VIII, 311 pages. 1982.

Vol. 137: International Symposium on Programming. Proceedings, 1982. Edited by M. Dezani-Ciancaglini and M. Montanari. VI, 406 pages. 1982.

Vol. 138: 6th Conference on Automated Deduction. Proceedings, 1982. Edited by D.W. Loveland. VII, 389 pages. 1982.

Vol. 139: J. Uhl, S. Drossopoulou, G. Persch, G. Goos, M. Dausmann, G. Winterstein, W. Kirchgässner, An Attribute Grammar for the Semantic Analysis of Ada. IX, 511 pages. 1982.

Vol. 140: Automata, Languages and programming. Edited by M. Nielsen and E.M. Schmidt. VII, 614 pages. 1982.

Vol. 141: U. Kastens, B. Hutt, E. Zimmermann, GAG: A Practical Compiler Generator. IV, 156 pages. 1982.

Vol. 142: Problems and Methodologies in Mathematical Software Production. Proceedings, 1980. Edited by P.C. Messina and A. Murli. VII, 271 pages. 1982.

Vol. 143: Operating Systems Engineering. Proceedings, 1980. Edited by M. Maekawa and L.A. Belady. VII, 465 pages. 1982.

Vol. 144: Computer Algebra. Proceedings, 1982. Edited by J. Calmet. XIV, 301 pages. 1982.

Vol. 145: Theoretical Computer Science. Proceedings, 1983. Edited by A.B. Cremers and H.P. Kriegel. X, 367 pages. 1982.

Vol. 146: Research and Development in Information Retrieval. Proceedings, 1982. Edited by G. Salton and H.-J. Schneider. IX, 311 pages. 1983.

Vol. 147: RIMS Symposia on Software Science and Engineering. Proceedings, 1982. Edited by E. Goto, I. Nakata, K. Furukawa, R. Nakajima, and A. Yonezawa. V. 232 pages. 1983.

Vol. 148: Logics of Programs and Their Applications. Proceedings, 1980. Edited by A. Salwicki. VI, 324 pages. 1983.

Vol. 149: Cryptography. Proceedings, 1982. Edited by T. Beth. VIII, 402 pages. 1983.

Vol. 150: Enduser Systems and Their Human Factors. Proceedings, 1983. Edited by A. Blaser and M. Zoeppritz. III, 138 pages. 1983.

Vol. 151: R. Piloty, M. Barbacci, D. Borrione, D. Dietmeyer, F. Hill, and P. Skelly, CONLAN Report. XII, 174 pages. 1983.

Vol. 152: Specification and Design of Software Systems. Proceedings, 1982. Edited by E. Knuth and E.J. Neuhold. V, 152 pages. 1983.

Vol. 153: Graph-Grammars and Their Application to Computer Science. Proceedings, 1982. Edited by H. Ehrig, M. Nagl, and G. Rozenberg. VII, 452 pages. 1983.

Vol. 154: Automata, Languages and Programming. Proceedings, 1983. Edited by J. Diaz. VIII, 734 pages. 1983.

Vol. 155: The Programming Language Ada. Reference Manual. Approved 17 February 1983. American National Standards Institute, Inc. ANSI/MIL-STD-1815A-1983. IX, 331 pages. 1983.

Vol. 156: M.H. Overmars, The Design of Dynamic Data Structures. VII, 181 pages. 1983.

Vol. 157: O. Østerby, Z. Zlatev, Direct Methods for Sparse Matrices. VIII, 127 pages. 1983.

Vol. 158: Foundations of Computation Theory. Proceedings, 1983. Edited by M. Karpinski, XI, 517 pages. 1983.

Vol. 159: CAAP'83. Proceedings, 1983. Edited by G. Ausiello and M. Protasi. VI, 416 pages. 1983.

Vol. 160: The IOTA Programming System. Edited by R. Nakajima and T. Yuasa. VII, 217 pages. 1983.

Vol. 161: DIANA, An Intermediate Language for Ada. Edited by G. Goos, W.A. Wulf, A. Evans, Jr. and K.J. Butler. VII, 201 pages. 1983.

Vol. 162: Computer Algebra. Proceedings, 1983. Edited by J.A. van Hulzen. XIII, 305 pages. 1983.

Vol. 163: VLSI Engineering. Proceedings. Edited by T.L. Kunii. VIII, 308 pages. 1984.

Vol. 164: Logics of Programs. Proceedings, 1983. Edited by E. Clarke and D. Kozen. VI, 528 pages. 1984.

Vol. 165: T.F. Coleman, Large Sparse Numerical Optimization. V, 105 pages. 1984.

Vol. 166: STACS 84. Symposium of Theoretical Aspects of Computer Science. Proceedings, 1984. Edited by M. Fontet and K. Mehlhorn. VI, 338 pages. 1984.

Vol. 167: International Symposium on Programming. Proceedings, 1984. Edited by C. Girault and M. Paul. VI, 262 pages. 1984.

Vol. 168: Methods and Tools for Computer Integrated Manufacturing. Edited by R. Dillmann and U. Rembold. XVI, 528 pages. 1984.

Vol. 169: Ch. Ronse, Feedback Shift Registers. II, 1-2, 145 pages. 1984.

Vol. 171: Logic and Machines: Decision Problems and Complexity. Proceedings, 1983. Edited by E. Börger, G. Hasenjaeger and D. Rödding. VI, 456 pages. 1984.

Vol. 172: Automata, Languages and Programming. Proceedings, 1984. Edited by J. Paredaens. VIII, 527 pages. 1984.

Vol. 173: Semantics of Data Types. Proceedings, 1984. Edited by G. Kahn, D.B. MacQueen and G. Plotkin. VI, 391 pages. 1984.

Vol. 174: EUROSAM 84. Proceedings, 1984. Edited by J. Fitch. XI, 396 pages. 1984.

Vol. 175: A. Thayse, P-Functions and Boolean Matrix Factorization, VII, 248 pages. 1984.

Vol. 176: Mathematical Foundations of Computer Science 1984. Proceedings, 1984. Edited by M.P. Chytil and V. Koubek. XI, 581 pages. 1984.

Vol. 177: Programming Languages and Their Definition. Edited by C.B. Jones. XXXII, 254 pages. 1984.

Vol. 178: Readings on Cognitive Ergonomics - Mind and Computers. Proceedings, 1984. Edited by G.C. van der Veer, M.J. Tauber, T.R.G. Green and P. Gorny. VI, 269 pages. 1984.

Vol. 179: V. Pan, How to Multiply Matrices Faster. XI, 212 pages. 1984.

Some Notation and Abbreviations

Notation	Meaning, Comments	Defined or First Used In Section(s) (see also Index)
A	algorithm	19,23
ar(A);ar(P)	number of arithmetical operations involved in A; required in order to solve a problem P	19,23
as(A)	number of additions/subtractions involved in A	32,33
AAPR	accumulation of the accelerating power via recursion	6
$b_Y(X,Y)$	bilinear form in X,Y	2
BA(n)	bilinear algorithm for nXn MM	2,22,23
BA(n,λ)	bilinear λ-algorithm for nXn MM	23
BBM	Boolean MM	18
bs(A)	bit-space used by A	23
bt(A)	bit-time used by A	23
bt(s),bt(*,s),bt(±,s)		18
bs(P),bt(P)	bit-time and bit-space of a computational problem P	23
C	the field of complex numbers	2
$c_\wp$	commutative rank	32
$cb_\wp$	commutative λ-rank	33
C(g,h)	g!/(h!(g-h)!)	8,9
cond	condition	25
D	domain of definition of problem or algorithm	Part 2 (Summary); 23
d	degree of λ-algorithm	6
d	shortest distance	18 only

det(W)	determinant of a matrix W	19
Det(n)	the problem of the evaluation of the determinant of an nXn matrix	Part 2 (Summary); 19
DFT(n)	discrete Fourier transform,	38,39
E	extension of a ring (field)	5
E,E(n),e(n),E(A,D,h), E(Z(V),D,h)	error bounds	Part 2 (Summary); 23-30
F	ring,field	2
F[λ]	ring of polynomials over F	6
f(i,j,q), f'(j,k,q), f''(k,i,q), f(α,q),f'(β,q), f''(γ,q)	constant coeficients (from F) of bilinear algorithms	2
f(i,j,q,λ), f'(j,k,q,λ), f''(k,i,q,λ), f(α,q,λ), f'(β,q,λ), f''(γ,q,λ)	coefficients (from F[λ]) of bilinear λ-algorithms	6
f,f',f'',f̃,f̃',f̃''		23
FFT	fast Fourier transform	Intr.,2,38
h(s)	2^{1-s}	23
u^H,W^H	complex conjugate of number u, conjugate transpose of matrix W	19
I (also I_n))	identity matrix (of size nXn)	19
ℓ_h(ℓ_2, ℓ_∞)	ℓ_h-norm of a matrix or of a vector	24
log u	logarithm to the base 2 of u	1

L_q, L_q'		2
L_q''		10
M	rank of algorithm, λ-rank of λ-algorithm	2,4,6
MA,MS	matrix addition,subtraction	20
MI	matrix inversion	Part 2 (Summary); 19
MM	matrix multiplication	Intr., 1
(m,n,p); also mXnXp MM	the problem of mXn by nXp MM	2
O(g(s)),o(g(s))	see Notation 18.1	Intr.,1,18
O, O_n	null matrix	19,20
PM	polynomial multiplication	2
Q	field of rational numbers	2
Q	unitary matrix (a QR-factor)	20
Q(s)	computed approximation to Q	26-30
$QR, \tilde{Q}R, QR^*$		20
R	upper triangular matrix (a QR-factor)	20
R(s)	computed approximation to R	26-30
R	field of real numbers	2
$\underline{R}$	set of vectors in the proof of Theorem 7.2	9 only
SLE	the problem of solving a system of linear equations	Part 2 (Summary); 19
sm(A)	number of scalar multiplications in A	32,33
T	trilinear form	10
TA	trilinear aggregating	Intr.,3,11

Tr(W)	trace of a matrix W	10
TMI	triangular matrix inversion	21
t	tensor	2,10
U,V,W,X,Y,Z	matrices	1,2,4,6,10
Z	ring of integers	2,5
Z(\)	ring of integers modulo \	2,5
Z(V)	output matrix	24-30
$\delta(i,j)$	$\delta(i,j)=0, i \neq j; \delta(i,i)=1$	2
Δ , Δ'	error value, error matrix	23-30
λ	see λ-algorithms	4,6
ρ , ρ_F	rank, rank over a ring F	2
$\rho(m,n,p)$	rank of mXnXp MM	2
$b\rho$	λ-rank	36
ω	exponent of MM	2
ω_F	exponent of MM over a ring F of constants	2
Σ,Π	symbols of sums, products	
Σ	diagonal matrix	20 only
$\lfloor u \rfloor$, $\lceil u \rceil$	see Notation 18.1	18
$\oplus$	direct sum of disjoint problems	8
Θ	direct sum of identical disjoint problems	2,5,8
$\boxtimes$	(tensor) product of bilinear problems	2,5,8
$\otimes$	direct (Kronecker) product of vectors, matrices, tensors, and of linear, bilinear, or polylinear forms	10,14,16

$\mathfrak{M}$	generalized MM	18 only
$\|\|\underline{v}\|\|, \|\|W\|\|$	norms of vector $\underline{v}$, matrix W	24
$t \leftarrow t'$	mapping (algorithm)	5,8
$\|S\|$	cardinality of a set S	
$\|u\|$	absolute value (modulus) of a number u	
$\subset$, $\underline{\subset}$	inclusion of one set into another	5
$\in$	inclusion of an element into a set	9
$\cup$	union of sets	5
■	end of clause, of proof, of algorithm	

CONTENTS

Introduction

Matrix multiplication (hereafter referred to as MM) is a basic operation of linear algebra, which has numerous applications to the theory and practice of computation. In particular, several important applications are due to the fact that MM is a substantial part of several successful algorithms for other computational problems of linear algebra and combinatorics, such as the solution of a system of linear equations, matrix inversion, the evaluation of the determinant of a matrix, Boolean MM, and the transitive closure of a graph. Moreover, the computational time required for MM is the dominating part of the total computational time required for all of those problems, that is, all such problems can be reduced to MM and can be solved fast if MM is solved fast.

How fast can we multiply matrices? The product of a pair of N X N matrices X and Y can be evaluated in a straightforward way using N^3 multiplications of the entries of X and Y and $N^3 - N^2$ additions of the resulting products. The best upper bound on the number of arithmetical operations for MM and for all related problems listed above remained $O(N^3)$ until 1968 while the best lower bounds on that number were of order N^2. The best lower bound is still of the same order N^2 but since 1968 the "upper" exponent 3 has been reduced to the currently record value 2.496. So far the latter progress made no impact on the practice of computing matrix products due to the substantial overhead of the asymptotic acceleration of MM. The impact on the theory of algorithms, however, was substantial. The mere existence of asymptotically fast algorithms for MM and for the related computational problems is encouraging. More important, a deeper insight into the subject of the design of efficient algorithms for arithmetical computations has been obtained due to the study of MM. Furthermore some specific techniques such as the duality method and the methods of the fast approximate computation first passed the tests of their efficiency in the study of MM and then were successfully applied to other computational problems. Also some rather unexpected theoretical applications followed. (For example, see Theorem 7.1 in Section 7, which implies a nontrivial fact of the theory of tensors, and note the quantitative measure of asynchronicity of linear computations defined in Section 39.)

The original intension of the present author was to summarize the knowledge accumulated through the years of the study of MM. In the process of writing the book, the author observed some new facts, developed some new insights, and arrived at some algorithms that promise some practical application.

The major subdivision of the material will be as follows. In Part 1 of this book the algorithms for MM will be devised that are currently asymptotically fastest, that is, they define the exponent 2.496. This will be an occasion to demonstrate all of the major techniques that historically have been applied to the

asymptotic acceleration of MM. In Part 2 the computations for several problems of combinatorial analysis and linear algebra will be reduced to MM and consequently accelerated. The traditional reduction in terms of the numbers of arithmetical operations will be extended to the reduction in terms of the numbers of bit-operations and bits of storage space involved. The bit-time and the bit-space used in the algorithms will be shown to be closely related to the values of the condition numbers of those algorithms; those condition numbers will be estimated. The bit-time complexity classification of the linear algebra problems rather unexpectedly differs from their arithmetical complexity classification. In Part 3 several algorithms for fast MM with smaller overhead will be presented. Those algorithms are superior over the asymptotically fastest algorithms for NXN MM for all sizes of interest, say, certainly for all $N < 10^{20}$. Furthermore, the unrestricted λ-algorithms defined in Section 40 promise to lead to an efficient practical method of computation in linear algebra. (That class of algorithms extends our study begun in [P80a] and [P82].) Also in Part 3 some well-known linear lower bounds on the complexity of algorithms of different classes will be derived for the sake of completeness and in order to demonstrate the basic active substitution method and some other customary lower bound techniques. The summaries introducing Parts 2 and 3, the titles of the sections and their first paragraphs may serve as the further guidance to the contents of the book.

To facilitate the selective reading from the book and its use for references, we will minimize the interconnections among Parts 1, 2 and 3 and even among the groups of subjects within each part, particularly within Part 3. (For instance, the readers may now try to read Section 39 on the asynchronicity with the occasional references to the preceding sections, if needed, or they may try to examine Examples 40.1-40.5 of the unrestricted λ-algorithms presented in the concluding Section 40.) On the other hand, we will unify our presentation, at least in Part 1, by using a model example (throughout Part 1) to illustrate the main ideas and the main techniques. (This line has clear historical parallels in the study of MM, which has been greatly influenced by the successful designs of algorithms in the form of modestly looking patterns.) For more selective reading about the asymptotic acceleration of MM, we refer to the papers [P84] (for a lighter presentation of that subject) and [P82c] and [P84a] (for a more concise treatment of it).

We will employ mostly algebraic techniques in Parts 1 and 3 and mostly the techniques of numerical analysis (in particular of error analysis and of numerical linear algebra) in Part 2.

The author tried to make the exposition elementary and basically self-contained. The outside results cited in Part 2 of the book will amount to the well-known estimates for the bit-complexity of the four arithmetical operations and

of the evaluation of square roots and to some elementary facts from the undergraduate level text books on numerical analysis. In Parts 1 and 3 a very modest amount of algebra (mostly just the concepts of the algebraic rings and fields) will be used. That amount can be substantially reduced if the readers assume that the computations have been performed with real or complex constants rather than over an arbitrary ring or field of constants and also if they skip the "harder" sections and remarks, such as our remarks on the correlation between the arithmetical complexity and the tensor rank (the concept of tensor rank is not used in our presentation except for the proof of isolated Theorem 36.2 in Section 36 but we indicate where that concept could be used in order to obtain a more comprehensive insight into the subject).

This monograph covers mostly the progress since 1978 that has not been covered in other books so far except for the excellent but very brief treatment in [K]. (Also the papers [P81],[S81], and [CW] remain very far from being complete and updated surveys on MM.) Several adjacent topics have already been well treated in the books and in the survey papers. We will omit those topics or only briefly mention about them referring the reader to [AHU], [BM], [K] on the design and analysis of algorithms for arithmetical computations, to [GvL] on the numerical aspects of linear algebra including the computations with sparse and special matrices; to [BGH] and [He] on parallel algebraic computing; to [DGK] on the pipe-line computation in linear algebra; and to the numerous publications on the computation with Toeplitz, Hankel, and circulant matrices and on the fast Fourier transform (FFT), see, in particular, [AHU], [B83], [BGY], [BM], [F71], [F72], [F72a], [FZ], [FMKL], [GvL], [K], [KKM], [Ra], [W80].

I will end this introduction with a brief overview of the history of MM and with my recollections of my own work on the subject. (In order to emphasize the more personal character of that overview, I will use the singular "I" rather than the plural "we" until the end of the introduction.)

Historically the study of MM was the second of the major subjects that have shaped the modern theory of the algeraic (arithmetical) computational complexity. The pioneering work of 1954 by A.M. Ostrowski, see [Os], introduced the first subject of that theory, that is, the problem of fast evaluation of polynomials; see [K] and [BM] for the history of the research on that subject where I was involved starting with 1959. In 1964 my former Ph.D. thesis advisor in Moscow University, Professor A.G. Vitushkin, suggested me reducing the mentioned earlier gap between the lower bound 2 and the upper bound 3 on the exponent in the case of a system of linear equations. This was a somewhat natural continuation of my thesis work, where I was fortunate to close the gap between the lower and the upper bounds on the number of arithmetical operations required for the evaluation of polynomials, see

[P62],[P64],[P66]. (Actually the basic active substitution techniques from [P62],[P64],[P66] turned out to be also useful for deriving some linear lower bounds on the arithmetical complexity of linear algebra problems, compare [BM], [K], [St72], [AS].) I have soon shifted to the study of MM and noted in December 1965 how NXN MM could be performed in about $N^3/2$ multiplications and in about $3N^3/2$ additions and subtractions, see Example 32.2 in Section 32. That result was a simple and almost straightforward extension of the previously developed techniques for the fast evaluation of polynomials with preconditioning (see [P66] or [K] for surveys) but my presentation of that result in January 1966 at the scientific seminar in Moscow headed by A.S. Kronrod, E.M. Landis, and G.M. Adel'son-Vel'skii was very well received. However, no interest elsewhere was raised, and the result remained unpublished until 1968 when it was rediscovered by S. Winograd, see [W68]. Furthermore, I could not get any support for my study of MM and had to interrupt that study until 1978. (I, however, published a short paper on that subject in 1972, see [P72].) Meanwhile MM became the subject of worldwide interest in 1968, when Volker Strassen discovered that the problem of MM and consequently several related computational problems of linear algebra can be solved in $O(N^{2.808})$ rather than in $O(N^3)$ arithmetical operations. Many scientists understood that discovery as a signal to attack the problem and to push the exponent further down. No further progress, however, followed for about 10 years. It happened that by that time, in 1978, I was able to come back to the study of MM. In September 1977 I started to work at the Department of Mathematics of the IBM Research Center, Yorktown Heights, New York, reporting the results of my work to S. Winograd, whose previous major publications [W67] and [W70] relied on [P66] and who later became the head of that department. In May 1978 I reexamined my algorithmic design published in 1972, improved it, and reduced the exponent first to 2.795 and soon thereafter to 2.781. That progress relied on the special techniques of trilinear aggregating (TA) introduced in my paper [P72] in 1972. TA turned out to be a general efficient method for the design of fast MM algorithms. The power of TA was substantially enhanced by combining it with two other important techniques introduced later (in 1978 and in 1979). Namely, fast algorithms for MM can be derived from the so called any precision aprpoximation algorithms (APA-algorithms) for Disjoint MM. The striking and ingenious idea of using APA-algorithms for the acceleration of MM was first introduced by D. Bini, M. Capovani, G. Lotti and F. Romani in [BCLR]. The efficiency of the application of that idea to MM was substantiated by D. Bini in [B80]. (We will describe the APA-algorithms in this book under the name λ-algorithms.) The idea of using Disjoint MM was due to A. Schonhage, see [S81]. The idea comes quite natural from the analysis of TA and of the so called direct sum problem. (The latter problem was well known in combinatorial analysis and in algebra. For the algebraic computation, that problem was first stated in the form of a conjecture by V. Strassen in 1973, see [St73].) The comparison of the applications of TA in [P72], [P78], [P80] and in the