VICTOR PAN – MY RESEARCH JOURNEY

Victor Y. Pan

Department of Mathematics and Computer Science
Lehman College - City University of New York
Bronx, New York 10468
Tel. (914) 737-2637 (home) and (718) 960-8568 (office)
Fax (718) 960-8969
E-Mail: victor.pan @ lehman.cuny.edu
Website: http://comet.lehman.cuny.edu/vpan/

1 MY MANIFESTO

I have been working in Mathematics, Computational Mathematics, and Computer Science for more than five decades, facing research challenges and seeking new insights and novel methods. I was thrilled whenever I discovered new keys that opened challenging scientific locks, particularly when *a single key opened a number of locks*, as this was the case with my techniques of active operation/linear substitution, trilinear aggregation, and transformation of matrix structures.

My work has contributed to the creation of the fields of the *Complexity of Algebraic Computations* and *Algebraic Multigrid* and to significantly advancing some other research areas such as *fast and processor-efficient parallel algorithms*, *computations with structured matrices* and *Symbolic-Numerical Computations*. Some of my techniques, insights, concepts and definitions are commonly used, sometimes as folklore.

My research has revealed a number of important but hidden links among apparently distant subjects and helped bring together research in various areas of computing such as symbolic computations, numerical computations, theoretical computer science and applied linear algebra.

I am grateful for recognition and support of my effort by leading experts, foundations (I received from National Science Foundation (NSF) over $2,500,000 for 1980-2020, over $1,000,000 of them for 2016-2020 and in 1993 I received ***Special Creativity Extension Award from the Numeric, Symbolic, and Geometric Computation Program of the CCR Division in the Directorate CISE of NSF***), journals, professional societies, research centers and universities (over $130,000 came from the City University of New York (CUNY) in 1989-2018). Enthusiastic reviews and citations of my work appeared in books, journals (including ***the Best Paper Award 2000, Journal of Complexity***, shared), and magazines. In 2013 I was designated a ***Fellow of American Mathematical Society "For Contributions to the Mathematical Theory of Computation"***.

According to Google Scholar and DBLP, I published four books (1623+LXXIV pages overall), over 20 surveys in journals and book chapters, over 170 research articles in journals and over 90 in refereed conference proceedings. Almost all my publications are in *Computer Science* and *Computational and Applied Mathematics*. I have also disseminated my findings through research reports, lectures at the universities, research centers and professional conferences worldwide, the Internet, and personal communication.

I published dozens of papers jointly with my current and former students. I advised and mentored 23 of them (including six African-Americans and three females) up to their PhD defenses, but some students continued joint research with me for more than a decade after the defense.

I have also served my fields as a Journal Editor, a Managing Guest Editor of Special Issues, a Member of the Program and Scientific Committees and a Local Chair of International Conferences and Workshops, where I also organized minisymposia.

2 Education and research areas

Next I will comment on my education and then on my research in ten major subject areas of Computer Science and Computational Mathematics. I will not cover my work of 1965–75 in Economics in the USSR and a number of my more sporadic research excursions into other areas. I will use the *acronyms* listed in Section 17, followed by the list of the references cited, and I will also refer to my works cited in the file of PUBLICATIONS (COMPLETE LIST). (My publications are also listed in the file of my SELECTED PUBLICATIONS and in Google Scholar and DBLP.)

My scientific destiny was decided in the 59th high school in Moscow, Russia, celebrated for having excellent teachers in mathematics. I was among many of its graduates who went to the famous MechMat Department of Moscow State University (MGU), headed by *Andrey Nikolaevich Kolmogorov*. He was one of the greatest mathematician of his time, and so was his student Vladimir Igorevich Arnold, also a graduate from the 59th school in Moscow.

My adviser *Anatoli Georgievich Vitushkin*, a world leading expert in the theory of functions of real and complex variables and a member of the Russian Academy of Sciences, was among Kolmogorov's distinguished disciples. He has also worked with a versatile scientist Alexander Semenovich Kronrod and like Kolmogorov and Kronrod had broad scientific interests.

From 1956 to 1961 I enjoyed learning Mathematics in MechMat of MGU. My first journal paper appeared in 1958 and was on the real function theory, but at that time Vitushkin guided me into *research in Computational Mathematics*, and from 1959 to 1964 almost all my publications as well as my PhD Thesis were in that field.

Having defended that thesis in 1964, I made living by working and publishing in Economics rather than mathematics because the job market in the USSR was quite restricted for people of Jewish ethnicity, like myself. In 1976 I emigrated to the USA and since 1977 have been working entirely in Computer Science and Computational Mathematics (see my employment history).

3 My first scientific breakthrough: polynomial evaluation

In 1962, by introducing a novel technique of active operation/linear substitution, I proved *optimality of the classical algorithm for polynomial evaluation*, commonly called Horner's. This gave positive answer to a question asked by Alexander Markowich Ostrowsky in 1955. Volker Strassen and Shmuel Winograd adopted my technique for proving the optimality of classical algorithms for some fundamental matrix computations (see Section 2.3 of [BM75]).

In that period I also accelerated polynomial evaluation by means of pre-processing.

My work has been surveyed in my paper [P66] and in the most fundamental Computer Science book [K81/97] by Donald E. Knuth, which cites my work and that of R.P. Brent most extensively among all its cited authors. The paper [P66] has made considerable impact in the West, leading to the emergence of the field of *Complexity of Algebraic Computations*.

I have become known in the West as *"polynomial Pan"*.

4 My second scientific breakthrough: fast matrix multiplication by means of trilinear decomposition and aggregation

Matrix multiplication (hereafter referred to as MM) is one of the central subjects of the theory and practice of computing, and the scientific world was tremendously impressed in 1969, when Strassen decreased the classical exponent 3 of MM to $\log_2 7 \approx 2.808$, that is, performed MM by using less than cubic time. In my book and review article in SIAM Review in 1984, both much cited at that time, I praised his discovery as well as his subsequent extensive work on algebraic computations, while he himself has been attracted to this field by my paper [P66] and has paid tribute to my work in his chapters, both called *"Pan's method"*, in [S72] and [S74].

Further progress toward performing MM in quadratic time was expected to come shortly, but all attempts to decrease the exponent 2.808 defied worldwide effort for almost a decade, until I decreased it in 1978.

My work was most widely recognized as a long-awaited breakthrough. The following excerpt from a letter by Donald E. Knuth is quoted here with his permission.

"I am convinced that his research on matrix multiplication was the most outstanding event in all of theoretical computer science during 1978. The problem he solved, to multiply $n \times n$ matrices with less than $O(n^{\log_2 7})$ operations, was not only a famous unsolved problem for many years, it also was worked on by all of the leading researchers in the field, worldwide. Pan's breakthrough was based on combination of brilliant ideas, and there is no telling what new avenues this will open."

Indeed my techniques prompted fast new progress in the field, with my participation. I have become widely known as *"matrix Pan"* and to the experts as *"matrix and polynomial Pan"*.

I devised my fast MM algorithms by means of

(i) reducing the bilinear problem of matrix multiplication to the equivalent problem of trilinear (tensor) decomposition and

(ii) nontrivially exploiting its cyclic symmetry in the case of matrix multiplication.

My combination of the two techniques, called *trilinear aggregation* in [P78], was not new in 1978, however, – it was introduced in my paper [P72] (in Russian). Its implementation in [P72] only supported an exponent below 2.85, but a better implementation in [P82] yielded an exponent below 2.7734. The paper [P72] was translated into English only in 2014, in arXiv:1411.1972, and was little known in the West until 1978.

Actually my trilinear aggregation technique of 1972 was a historic landmark on a wider scale. *It produced the first nontrivial decomposition of a tensor and the associated trilinear form that defined a new efficient algorithm for matrix computations.* Subsequently tensor decomposition has become a popular tool for devising highly efficient matrix algorithms in many areas of scientific computing. Says E.E. Tyrtyshnikov, a renowned expert in tensor decomposition:

“We should be especially thankful to Victor Pan for the link between the bilinear algorithms and trilinear tensor decompositions. Although it looks simple and even might be regarded as a folklore by now, this observation still has its creator, and by all means and for all I know it is due to the work of Victor Pan written in the Soviet period of his life.”

Since 1978 my trilinear aggregation has been routinely employed by myself and my successors for devising new fast MM algorithms. After the stalemate from 1969 to 1978 the MM exponent was decreased several times in 1979–1981, then again twice in 1986, reaching the record value 2.376 in [CW86/90]. It was decreased again in 2010–
2014, but only nominally. Every decrease relied on amazing novel techniques built on the top of the previous ones, always employing the reduction of the MM problem to trilinear aggregation, as has been pointed out on page 255 of the celebrated paper [CW86/90].

As Arnold Schönhage has written at the end of the introduction of his seminal paper [S81], however, *all these exponents of MM have been just "of theoretical interest". They hold only for inputs "beyond any practical size", and "Pan's estimates of 1978 for moderate" input sizes were "still unbeaten".* Actually in [P79], [P80], [P81] and [P82], I successively decreased my record exponent for feasible MM (that is, for MM of moderate sizes $n \times n$, say, up to $n \leq 1,000,000,000$). My exponent of [P82], below 2.7734, still remains the record by the end of 2017. All smaller exponents have been obtained by ignoring the *curse of recursion*, that is, by means of applying long recursive processes, each squaring the input size. The resulting algorithms beat the classical one only for inputs of immense sizes.

My algorithms promise to be highly efficient in practice: the implementations by Igor Kaporin of an algorithm from [P84a] in [K99] and of that of [LPS92] in [K04] use substantially smaller computer memory and are more stable numerically than Strassen's algorithm.

I surveyed the progress up to the date in a book [P84b] and long papers [P84a] and [P17]. In 1984 I focused on the decrease of the exponent of MM because this was the focus of the research community, but in 2017 I paid much more attention to the acceleration of feasible MM.

In [HP98], jointly with my student Xiaohan Huang, I accelerated rectangular MM, which implied new record complexity estimates for the computations of the composition and factorization of univariate polynomials over finite fields.

5 Hierarchical aggregation as a springboard for the Algebraic Multigrid (1980). Compact Multigrid (1990–1993)

In [MP80], jointly with Miranker, I introduced hierarchical aggregation/disaggregation processes, substantially responsible for the emergence of the popular field of *Algebraic Multigrid*.

Jointly with Reif, in SPAA 1990, SIAM J. of Scientific and Statistical Computing 1992 and CAMWA 1990 and 1993, I proposed a simple but novel acceleration technique of *Compact Multigrid*.

6 Parallel algebraic and graph algorithms (1985–2001)

Throughout the years of 1985–2001, prompted by high recognition of my joint paper with John H. Reif at STOC 1985, I proposed, both with coauthors and by myself, a variety of new efficient parallel algorithms and in particular a number of fast and processor-efficient parallel algorithms for computations with matrices, polynomials, and graphs. They relied on our novel nontrivial techniques, and I regularly presented them at the most competitive conferences in this field such as ACM STOC, IEEE FOCS, ICALP and ACM-SIAM SODA and published them in leading journals such as SICOMP, JCSS, Algorithmica and Information and Computation.

a) *Fast and processor efficient algorithms for matrix and polynomial computations.* In STOC 1985, jointly with Reif I introduced fast and processor efficient parallel algorithms for the solution of dense and sparse linear systems of equations. The study of processor efficiency of fast parallel algorithms was a novelty at that time. In 1985–86 this work was covered *in the magazines Science, Science News, and Byte.* The algorithm for sparse linear systems of equations has been implemented *on the supercomputers of NASA and Thinking Machines Corp.* By myself and jointly with coauthors I continued working on parallel matrix and polynomial computations for more than a decade. We proposed nontrivial novel techniques, extended the list of the known fast and processor efficient parallel algorithms, and improved the known complexity bounds for the following fundamental computational problems: (i) the solution of general and structured linear systems of equations with integer input (see my papers in TCS 1987, IPL 1989, and SICOMP 2000) and over abstract fields (see my paper in CAMWA 1992 and my joint papers with Bini and Gemignani in ICALP 1991 and Kaltofen in SPAA 1991 and FOCS 1992), (ii) the computation of polynomial greatest common divisors (GCDs), least common multiples, and Padé approximations (see my papers in CAMWA 1992 and TCS 1996), (iii) polynomial division (see my joint papers with Bini in J. of Complexity 1986, FOCS 1992, and SICOMP 1993), and (iv) the computation of the determinant, the characteristic polynomial, and the inverse of a matrix (see my joint papers with Galil in IPL 1989 and Huang in J. of Complexity 1998).

b) *Graph algorithms.* To obtain fast and processor efficient parallel algorithms for the computation of matchings and paths in graphs, I have extended the known nontrivial reductions to matrix computations and applied some novel techniques for these computations. I published these results in FOCS 1985 and Combinatorica 1988 jointly with Galil, in JCSS 1989, IPL 1991, and SICOMP 1993 jointly with Reif, in SICOMP 1995 jointly with Preparata, in Algorithmica of 1997 jointly with Han and Reif, and in my own chapter in the Handbook on Computer Science of 1993.

c) In my joint works with Shallcross and my student Lin–Kriz, I proved the *NC-equivalence* of the integer GCD and planar integer linear programming problems, which was a well-known theoretical challenge (see our papers in SODA 1992, FOCS 1993 and SICOMP 1998).

7 Univariate polynomial root-finding (1985–2017). Nearly optimal solution of a four millennia old problem

Univariate polynomial root-finding has been central in mathematics and applied mathematics for four millennia. It was studied already on Sumerian clay tablets and Egyptian papyrus scrolls but has modern applications to signal processing, financial mathematics, control theory, computational algebraic geometry and geometric modeling.

Hundreds of efficient algorithms have been proposed for its solution. Two-part book published with Elsevier, by J.M. McNamee in 2007 (354 pages) and jointly by J.M. McNamee and myself in 2013 (728 pages), covers nearly all of them up to the date, in a *unique comprehensive coverage of this popular subject area.*

Since 1985 I have been doing research in that area and in the related areas of computation of approximate polynomial GCDs, matrix eigenvalues and eigenvectors, and the solution of a system of multivariate polynomial equations. Next I briefly outline some of my results referring the reader for further information to my papers cited below in parts (a)–(g) and the papers (individual and joint with my students) in FOCS 1985 and 1987, CAMWA 1985, 1987, 1995, 1996, 2011 (two papers), and 2012 (two papers, one of them joint with McNamee), SICOMP 1994, J. of Complexity 1996 and 2000 (four papers), JSC 1996, ISSAC 2010 and 2011, and SNC 2011 and 2014 (two papers).

a) In STOC 1995 (and also in CAMWA 1996, ISSAC 2001, and JSC 2002) I combined the advanced techniques by Schönhage and by Neff and Reif with my novelties in exploiting the geometry of the complex plane, precision control by using Padé approximation, and recursive lifting and descending. As a result I have substantially accelerated the known algorithms. My resulting divide-and-conquer algorithms of STOC 1995 approximate all roots of a univariate polynomial nearly as fast as one can access the input coefficients, in *optimal* (up to a polylogarithmic factor) *Boolean time*. I have surveyed my work up to the date in SIAM Review 1997 and more informally in American Scientist 1998. I cover it in more detail in JSC 2002 and Chapter 15 of my book of 2013, joint with McNamee and already cited.

(b) Hermann Weyl's Quad-tree construction of 1924 enables the solution of a univariate polynomial equation in roughly quartic arithmetic time. James Renegar decreased the time bound to cubic in 1987, and I reached quadratic arithmetic time bound in J. of Complexity 2000. Most of the computations of my algorithm require low precision, and this suggested that *nearly optimal Boolean time can be also reached* based on extension or refinement of this algorithm. I have not pursued that goal, but very recently Ruben Becker, Michael Sagraloff, Vikram Sharma and Chee Yap achieved this (see [BSSY18]). Their work boosted interest to that direction because the approach promises to be practically more efficient than the divide-and-conquer method.

c) *Approximation of the real roots* of a polynomial is an important goal because in many applications, e.g., to algebraic optimization, only r real roots are of interest and because frequently they are much less numerous than all n complex roots. My algorithms (joint with my students) in SNC 2007, CAMWA 2011, CASC 2012 and 2014, TCS 2017 accelerate the known algorithms for this problem by a factor of n/r.

d) My algorithm in ISSAC 2013 and JCS 2016 (joint with Elias P. Tsigaridas) is nearly optimal for a more narrow goal of real *polynomial root-refining*, rather than root-finding. Likewise my algorithm (also joint with Elias P. Tsigaridas) at SNC 2014 and TCS 2017 refines all complex roots at a nearly optimal Boolean complexity bound.

d) Together with Bini in J. of Complexity 1996, with Bini and Gemignani in CAMWA 2004, ETNA 2004 and Numerische Mathematik 2005, with McNamee in CAMWA 2012, by myself in CAMWA 2005, and jointly with my present and former students in ISSAC 2010, CAMWA 2011, LAA 2011, CASC 2012, SNC 2014, TCS 2017 and a chapter in the SNC volume of 2007, published by Birkhäuser, I proposed *novel matrix methods for polynomial root-finding*. Unlike many previous companion matrix methods, we preserve and exploit the structure of the associated companion and generalized companion matrices, and yield numerically stable solution, while keeping the arithmetic cost at a low level.

I further extended these algorithms to the solution of the eigenproblem for a general matrix in SODA 2005 and CAMWA 2006 and 2008.

e) Jointly with Bini, I proposed and elaborated upon an algorithm that *approximates all eigenvalues of a real symmetric tridiagonal matrix by using nearly optimal Boolean time*. This is a popular and important problem of matrix computations. We proposed the first algorithm of this kind, presented in some detail in SODA 1991 and then in Computing 1992 and SICOMP 1998.

f) Computation of *approximate polynomial GCDs* has important applications in control and signal processing. My papers in SODA 1998 and Information and Computation of 2001 yielded a new insight into this computational problem by exploiting its links to polynomial root-finding, matching in a graph, and Padé approximation.

8 A system of multivariate polynomial equations. Best Paper Award (1996-2005)

My joint papers with Bernard Mourrain in Calcolo 1996, STOC 1998 and J. of Complexity *(Best Paper Award for 2000)*, with Didier Bondifalat and Bernard Mourrain in ISSAC 1998 and LAA 2000, with Bernard Mourrain and Olivier Ruatta in SICOMP 2003, and with Ioannis Z. Emiris in ISSAC 1997, JSC 2002, CASC 2003, and J. of Complexity 2005, and in the references therein, introduced and analyzed a number of novel and now popular techniques and algorithms for the approximation of the roots of dense and sparse *systems of multivariate polynomials.* The algorithms exploits the structure of the associated matrices.

9 Matrix structures: benefits and unification (1987–2017)

This area is highly important for both theory and practice of computing. It was studied already in the 19th century and with increased intensity in the recent decades because of importance of this work for handling big data.

My contributions can be traced back to 1987 and include the results in the following directions, besides the applications to polynomial root-finding, already cited.

a) *Unification of structured matrix computations by using their displacement representation and the transformation of matrix structures.* The four most popular matrix structures of Toeplitz, Hankel, Vandermonde, and Cauchy types have different features, which allow different computational benefits. In particular, the Cauchy matrix structure, unlike the three other ones, is invariant in both row and column interchange and allows approximation by rank structured matrices, which can be highly efficiently handled by means of the Fast Multipole Method.

The matrices of all four classes share, however, an important feature: they can be represented in compressed form through their displacements of small rank. Every matrix M can be expressed via its displacements $AM - MB$ and $M - AMB$ under mild restriction on operator matrices A and B, and for each of the four classes of structured matrices and a proper pair of operators of shift and/or diagonal scaling the displacement has small rank and thus can be represented with fewer parameters (typically with $O(n)$ parameters for an $n \times n$) structured matrix, having n^2 entries. By using advanced techniques one should extend this representation to yield dramatic saving of computer memory and time for computations with such matrices.

The approach was proposed in [KKM79] by Kailath, Kung and Morf, who demonstrated its power by multiplying by a vector an $n \times n$ Toeplitz-like matrix (having structure of Toeplitz type) by using $O(n)$ memory and $Q(n \log n)$ flops (arithmetic operations). The MBA divide-and-conquer algorithm of 1980 of Morf and of Bitmead and Anderson has extended this KKM 1979 progress to the inversion of Toeplitz-like matrices and the solution of Toeplitz-like linear system of equations, and the natural challenge was the extension to computations with the important matrix classes having other structures.

I contributed to further progress with my two books – of 1994 (with Bini) and 2001 – and dozens of papers by myself and joint with coauthors.

In ISSAC 1989 and MC 1990, I unified fast computations with the four listed matrix classes in a rather unexpected way. Namely, I observed that one can transform matrix structure at will by transforming the associated operator matrices, and moreover can do this by multiplying a given structured matrix by Hankel and Vandermonde matrices and their transposes. By applying such transformations of matrix structures one can *extend any successful algorithm for the inversion of the structured matrices of any of the four classes to the inversion of the matrices of the three other classes, and similarly for solving linear systems of equations.*

One can always use the simple reversion matrix as a Hankel multiplier and frequently can use the matrix of the discrete Fourier transform or its Hermitian transpose as a Vandermonde multiplier. In some cases such transformations enable dramatic improvement of the known algorithms.

For example, by 1989 cubic time was required for the inversion of Cauchy-like matrices and likewise for the Nevanlinna–Pick fundamental problem of rational approximation, closely linked to this task. *My transformations immediately decreased the known cubic upper bounds on the time-complexity of these highly important computational problems to nearly linear.*

Unlike the multiplication algorithm of [KKM79], the MBA inversion algorithm is numerically unstable, and this limits applications of the latter recipe. Later, however, *my approach has become basic for a stream of highly efficient practical numerical algorithms* for Toeplitz linear systems of equations: the algorithms begin computations with the converse reduction to the Cauchy-like case and then exploit either the invariance of Cauchy structure in row and column interchange (cf. [GKO95]) or the link of this structure to the *rank structure* of matrices and consequently to the Fast Multipole Method (cf. [CGS07], [MRT05], [XXG12]). In view of of such a link one is challenged to extend my approach to unification of computations with matrices having displacement and rank structures, which could be highly important for both theory and practice of matrix computations. Very recent progress towards meeting this unification challenge was reported in [BT17].

In 2013–2017 I extended my method to Vandermonde and Cauchy matrix-by-vector multiplication, the solution of Vandermonde and Cauchy linear systems of equations, and polynomial and rational interpolation and multipoint evaluation. For all these classical problems, the known numerical algorithms, running with bounded precision (e.g., the IEEE standard double precision), required quadratic arithmetic time, and I decreased it to nearly linear (see my papers in CASC 2013, LAA 2015 and MC, in press).

For another application of my techniques, in [P16] I formally supported empirical observation of many researchers (which *remained with no proof for decades*) that a Vandermonde matrix is ill-conditioned (that is, close to singular) unless it is close (up to scaling) to the matrix of discrete Fourier transform, whose knots are nearly equally spaced on or near the unit circle centered in the origin.

b) For *alternative and more direct unification* of computations with structured matrices of the four classes, one can express them in terms of operations with the displacements. The MBA algorithm of 1980 does this for Toeplitz-like matrices. I extend it to Cauchy-like matrices first jointly with my student Ai-Long Zheng in LAA 2000 (submitted in 1996) and then jointly with Vadim Olshevsky in FOCS 1998. In SODA 2000 and in chapter 5 of my book of 2001 I extended the MBA algorithm in a unified way for computations with various structured matrices.

c) *Efficient algorithms for structured matrices and links to polynomial and rational computations.* In SIAM Review 1992, CAMWA 1992, 1993 (jointly with my students), TCS 1996, and Annals of Numerical Mathematics 1997 (by myself), and ICALP 1999, jointly with Vadim Olshevsky, I presented new efficient algorithms for various fundamental computations with structured matrices such as computing their ranks, characteristic and minimum polynomials, bases for their null spaces, and the solutions of structured linear systems of equations. Furthermore I have also extended successful methods for computations with structured matrices to some fundamental computations with polynomials and rational functions. Conversely, in SNC 2014 and TCS 2017, jointly with Elias P. Tsigaridas, I deduced nearly optimal estimates for the Boolean complexity of some fundamental computations with Vandermonde and Cauchy matrices by reducing these computations to the ones for polynomials and modifying the known fast algorithms for the latter problems.

10 Newton's iteration for general and structured matrix inversion

Newton's iteration reduces matrix inversion to matrix multiplications, and this is valuable for various reasons, in particular for parallel computations. My paper with R. Schreiber in SISSC 1991 works out nontrivial initialization policies as well as some variations of the iteration that enhance performance. In Chapter 6 of my book of 2001 and in my paper with my students in MC 2006 I further improved performance by applying *homotopy continuation* techniques.

The iteration is attractive for the inversion of structured matrices, provided that the structure is preserved in iterative process. In J. of Complexity 1992, IEEE Transaction on Parellel and Distributed Systems 1993, and SIMAX 1993 I achieved this by means of *recursive re-compression*, that is, by recursively compressing displacements. This enabled *superfast solution algorithms*, running in nearly linear arithmetic time. The resulting algorithms allow processor efficient parallel implementation, and I have unified them over various classes of structured matrices. I presented these results in Chapter 6 of my book of 2001, my paper of 2010 in Matrix Methods: Theory, Algorithms and Applications, and with coauthors in LAA 2002, TCS 2004, Numerical Algorithms 2004, and MC 2006.

11 Computation of the determinant of a matrix

This classical problem has important applications in modern computing, e.g., to the computation of *convex hulls* and *resultants*, with further link to the solution of multivariate polynomial systems of equations.

a) In TCS 1987 (Appendix) and IPL 1988 I reduced the computation of the determinant of a matrix to the solution of linear systems of equations and then applied *p-adic lifting* to yield the solution efficiently. By extending this approach John Abbott, Manuel Bronstein and Thom Manders in ISSAC 1999, Wayne Eberly, Mark Giesbrecht and Gilles Villard in FOCS 2000, and myself jointly with Ioannis Z. Emiris in JSC 2003 obtained some of the most efficient known symbolic algorithms for the computation of the determinant of a matrix and the resultant of a polynomial system.

b) I published novel algorithms for computing determinants in TCS 1999, jointly with three coauthors from INRIA, France, and in Algorithmica of 2001, jointly with my student Yanqiang Yu. The algorithms perform computations with single or double IEEE standard precision, based on algebraic techniques (in the TCS paper) and on numerical techniques (in the Algorithmica paper), use small arithmetic time, and certify the output. The TCS paper has accelerated the computations by means of output sensitive and randomization methods, novel in this context.

12 Synergy of symbolic and numerical computations

Numerical and symbolic algorithms are the backbone of modern computations for Sciences, Engineering, and Signal and Image Processing, but historically these two subject areas have been developed quite independently of one another. Combination of their techniques can be highly beneficial.

Since the early 1990s I have been promoting such benefits as an organizer of conferences, as a member of their Program Committees and as the Managing Editor of four Special Issues of the Theoretical Computer Science (TCS) on this subject in 2004, 2008, 2011 and 2013. Perhaps even stronger impact into this direction was from my books of 1994 (joint with Dario Bini), 2001 (by myself), and 2013 (joint with John M. McNamee), and from my surveys in SIAM Review 1992 and 1997, in NATO ASI Series published by Springer 1991, Academic Press 1992, and Kluwer 1998, in the electronic proceeding of IMACS/ACA 1998, and in my chapters (with co-authors) in four Handbooks of 1999, 2004, 2009 and 2014, as well as from dozens of my research papers. For example, I have not edited a Special Issue of TCS on Symbolic-Numerical Algorithms in 2017, but the Issue published my two joint papers with Elias P. Tsigaridas and my joint paper with my student Liang Zhao, out of my four my joint papers submitted and 13 papers overall in the Issue.

13 Randomized preprocessing (2007–2017): added chaos stabilizes fundamental numerical matrix computations

Since 2007 I have been working on randomized pre-processing of matrix computations. I have contributed a new direction, new insight, and novel techniques to the popular area of randomized matrix computations. See my papers (some joint with my students) in SNC 2007 (two papers) and 2009, CSR 2008, 2010, and 2016, TCS 2008, CAMWA 2009, LAA of 2009, 2010 (two papers), 2011, 2012, 2013, 2015 and 2017 (two papers), ISSAC 2011, CASC 2015, and report in arXiv: 1611.01391.

I have advanced the known numerical algorithms for both nonsingular and homogeneous singular linear systems of equations. In particular I proved that, with a probability near one, randomized multiplicative preprocessing numerically stabilizes Gaussian elimination with no pivoting (GENP) and block Gaussian elimination, and I obtained similar results for any nonsingular and well-conditioned (possibly sparse and structured) multiplicative pre-processors and for the average input. This should embolden the search for new efficient sparse and structured multipliers, and I proposed some new classes of them. My work on this subject with my students appeared in TCS 2008, LAA 2012, LAA 2013, LAA 2015 and LAA 2017. GENP with randomized pre-processing should be practically valuable because pivoting (row/column interchange) is communication intensive and because Gaussian elimination is most used algorithm in matrix computations. Some implementation of GENP applied to an input pre-processed with ad hoc random multipliers appeared in a series of papers by Mark Baboulin et al. beginning in 2012. My study should help refine such implementations and provide formal support for this approach.

14 Superfast and accurate low rank approximation

By extending my randomization techniques I obtained substantial progress in the highly popular area of low rank approximation (hereafter referred to as *LRA*) of a matrix. This is one of the central problems of modern computing because of its highly important applications to numerical linear algebra, machine learning, neural networks, and data mining and analysis. In CSR 2016 (jointly with my student Liang Zhao) I proposed a new insight into this subject, provided formal support for the empirical power of various known sparse and structured multipliers and defined some new classes of efficient multipliers.

In arXiv:1611.01391 of November 2016, jointly with my students, I studied computation of LRA by using much fewer flops and memory cells than the input matrix has entries. I call such algorithms *superfast* assuming by default that they are also superefficient in using memory.

We first showed that any superfast algorithm fails to compute accurate LRA of the worst case input. Then, however, we proved that with a high probability the well-known cross-approximation algorithms (as well as some even more primitive algorithms) compute accurate LRAs (i) of a random input matrix allowing LRA, (ii) of the average input matrices allowing LRA and (iii) even of any input matrix allowing LRA and pre-processed with a Gaussian random multiplier. We proved our results in various ways, providing insights from different angles and allowing various error and probability estimates.

Since we proved that superfast LRAs fail for the worst case input, our pre-processing cannot be superfast, but empirically in our tests with benchmark inputs coming from discretized PDEs and some other real world input data, superfast pre-processing with our sparse and structured multipliers was consistently as efficient as with Gaussian random multipliers.

In our study we introduced various new insights and novel techniques for LRA and *achieved synergy* by combining some LRA techniques developed by two groups of researchers in Numerical Linear Algebra and Computer Science independently of one another.

15 Dramatic acceleration of celebrated matrix algorithms

In arXiv:1611.01391 we presented two surprising applications of low-rank approximation – we *dramatically accelerated the Conjugate Gradient (CG) algorithms* and the bottleneck stage of computing generators for applications of *Fast Multipoint Methods (FMM)*. In some applications these generators are given from outside, but more they typically must be computed, e.g., they must be computed superfast in order to support our acceleration of CG algorithms. Both FMM and CG algorithms are celebrated for being among the most important algorithms of the 20th century (see [C00] and [GL13]).

16 Concluding remarks

Throughout my career my work has advanced the state of the art of various fundamental subjects of Computational Mathematics and Computer Science such as computations with general and structured matrices, polynomials, integers and graphs, for example, polynomial evaluation, interpolation, division, factorization, and root-finding, solution of general and structured linear systems of equations, computation of linear recurrences, matching and paths in graphs, the sign and the value of the determinant of a matrix.

While devising new efficient algorithms, I proposed novel techniques and new insights and revealed hidden links among various subject areas and computational problems, e.g., (i) between matrix multiplication and tensor decomposition, (ii) among matrices with various structures, and (iii) between Fast Multiplole method and Conjugate Gradient algorithms. This list can be readily extended.

My novel ideas and techniques have been well recognized, but sometimes not immediately: each of my trilinear aggregation of 1972 and my transformation of matrix structures of 1989 waited for six years before they became widely known and appreciated, the value of my contribution of 2000 to the quadtree root-finding is only now becoming recognized, and similar story may very well occur with my novel insights into and techniques for low-rank approximation as well. In such cases the contributions are appreciated and praised more than their author, who is rewarded just by witnessing the resulting progress in the field.

I have been partly responsible for the creation of the fields of *Algebraic Complexity of Computations*. My long survey [P66] inspired the interest of Volker Strassen, Shmuel Winograd and many other renowned researchers, which has led to creation of the former field.[1] Their work in turn attracted me to this field again. For another striking example of *cross-fertilization*, my renewed interest to this field was prompted by the concise but far-fetching and clear exposition in the cited book [BM75] by Allan Borodin and Ian Munro, which was the first book in Math in English that I have read after moving to the USA in 1977. In 1979 I learned from Borodin that his interest to the field was inspired by my paper [P66] and Strassen's [S69].

My paper with Willard L. Miranker [MP80] was the beginning of the field of the *Algebraic Multigrid*, now popular.

My survey in SIAM Review in 1992 and my book with Dario Bini, published by Birkhäuser in 1994 as well as dozens of my research papers (individual and joint with Dario Bini and later with my students) have demonstrated various benefits of combining Symbolic and Numerical techniques, achieving synergy and contributing to the emerging field of *Symbolic and Numerical Computations*.[2]

My book with Dario Bini (1994) is called *"Polynomial and Matrix Computations"* and includes a number of new research results by the authors. It covers its title subjects both thoroughly and comprehensively according to its reviews (see some excerpts below) and was frequently cited, as well as my three other books (also devoted to polynomial and matrix computations) and my surveys in SIAM Review on matrix multiplication (1984), polynomial and matrix computations (1992), and polynomial root-finding (1997).

Excerpts from *SIGACT News*, ACM Press, 26, 2, pages 26–27, June 1995: "We are now greeted with the release of a book covering the basic, foundational material of the

[1]The next and seminal paper [W67] in this subject area begins with: "Introduction.-In reference [1], V. Ya. Pan summarized the results about the minimum number of multiplications and additions required to compute a polynomial. In particular, Pan proved that the minimum number of multiplications/divisions required to compute $P_n(x)x = a_0 + a_lx + \cdots + a_nx^n$ is n. The theorem of this note includes this result of Pan's as a special case, and also shows that the minimum number of multiplications/divisions required to compute the product of an $n \times n$ matrix by a vector is $m \cdot n$."

[2]Besides consistently producing papers on the subjects of this field (the latest three in the TCS 2017), I was the Corresponding and Managing Guest Editor of four special issues of the TCS on its subjects in 2004, 2008, 2011, and 2013.

algebraic algorithm field, written by the authors who are leading researchers in the field and are responsible for many of the current best algorithms...... For researchers in the field of algebraic algorithms, this is a "must-have" book, both as a reference and the review of basic material...... In conclusion, for researchers in the field of algebraic computing, I highly recommend this book as an essential addition to your bookshelf."

Excerpts from *SIGSAM Bulletin,* ACM Press, 30, 3, pages 21–23, September 1996: "The book covers an impressive range of algorithmic issues in Theoretical Computer Science, Symbolic Computer Algebra and Numerical Computation, and the presence of several latest methods and results makes it exciting to read. It would be useful to a specialist in any of the above areas who wishes to undergo a rigorous study of polynomial or matrix operations for large problems using exact or approximate arithmetic...... The book is outstanding...... In conclusion, the book by Bini and Pan is an excellent companion for researchers and advanced students. Given, moreover, that it is a handy reference book, it should be present in every good library."

17 Acronyms

"CACS" stands for "Proceedings of Conference on Applications of Computer Algebra"

"CAMWA" stands for "Computers and Mathematics (with Applications)"

"CSR" stands for "Proceedings of Computer Science in Russia"

"FOCS" stands for "Proceedings of IEEE Symposium on Foundations of Computer Science"

"ICALP" stands for "Proceedings of International Colloquium on Automata Languages and Programming"

"IPL" stands for "Information Processing Letters"

"ISSAC" stands for "Proceedings of ACM International Symposium on Symbolic and Algebraic Computation"

"JCSS" stands for "Journal of Computer and System Sciences"

"JSC" stands for "Journal of Symbolic Computation"

"LAA" stands for "Linear Algebra and Its Applications"

"LNCS" stands for "Lecture Notes in Computer Science"

"MC" stands for "Mathematics of Computation"

"SICOMP" stands for "SIAM Journal on Computing"

"SIMAX" stands for "SIAM Journal on Matrix Analysis and Applications"

"SNC" stands for "Symbolic-Numerical Computations or Proceedings of Workshop on SNC"

"SODA" stands for "Proceedings of ACM-SIAM Symposium on Discrete Algorithms"

"SPAA" stands for "Proceedings of ACM Symposium on Parallel Algorithms and Architecture"

"STOC" stands for "Proceedings of ACM Symposium on Theory of Computing"

"TCS" stands for "Theoretical Computer Science"

References

[BM75] Allan Borodin and Ian Munro, The Computational Complexity of Algebraic and Numeric Problems, American Elsevier, NY, 1975.

[BSSY18] Ruben Becker, Michael Sagraloff, Vikram Sharma and Chee Yap, A Near-Optimal Subdivision Algorithm for Complex Root Isolation based on the Pellet Test and Newton Iteration, Journal of Symbolic Computation, 86, 51–96, May–June 2018.

[BT17] Bernhard Beckermann and Alex Townsend, On the singular values of matrices with displacement structure, SIAM J. on Matrix Analysis and Applications, 2017.

[C00] Barry A. Cipra, The Best of the 20th Century: Editors Name Top 10 Algorithms, SIAM News, 33, 4, 2, May 16, 2000.

[CGS07] S. Chandrasekaran, M. Gu, X. Sun, J. Xia, J. Zhu, A Superfast Algorithm for Toeplitz Systems of Linear Equations, SIAM. J. on Matrix Analysis and Applications, 29, 4, 1247–1266, 2007.

[CW90/86] Don Coppersmith, Shmuel Winograd, Matrix Multiplicaton via Arithmetic Progressions, J. of Symbolic Computations, 9, 3, 251–280, 1990. Proc. version in 19th ACM Symposium on Theory of Computing (STOC 1987), 1–6, ACM Press, New York, NY, 1987. Also Research Report RC 12104, IBM T.J. Watson Research Center, August 1986.

[GKO95] Israel Gohberg, Thomas Kailath and Vadim Olshevsky, Fast Gaussian Elimination with Partial Pivoting for Matrices with Displacement Structure, Mathematics of Computation, 64, 1557–1576, 1995.

[GL13] Gene H. Golub, Cleve F. Van Loan, Matrix Computations, The Johns Hopkins University Press, Baltimore, Maryland, 2013 (fourth edition).

[HP97] Xiohan Huang, Victor Y. Pan, Fast Rectangular Matrix Multiplication and Applications. Journal of Complexity, 14, 2, 257–299, 1998. Proc. version in Proc. of ACM International Symposium on Parallel Algebraic and Symbolic Computation (PASCO'97), 11–23, ACM Press, NY, 1997.

[K81/97] Donald E. Knuth, The Art of Computer Programming, volume 2, Addison-Wesley, 1981 (second edition), 1997 (third edition).

[K99] Igor Kaporin, A Practical Algorithm for Faster Matrix Multiplication, Numerical Linear Algebra with Applications, 6, 8, 687–700, 1999.

[K04] Igor Kaporin, The Aggregation and Cancellation Techniques as a Practical Tool for Faster Matrix Multiplication, Theoretical Computer Science, 315, 2–3, 469–510, 2004.

[KKM78] Thomas Kailath, Sun-Yuan Kung, and Martin Morf, Displacement Ranks of Matrices and Linear Equations, J. of Mathematical Analysis and Applications, 68, 395–407, 1979. See also Thomas Kailath, Sun-Yuan Kung, and Martin Morf, Displacement Ranks of Matrices and Linear Equations, American Mathematical Society (N.S.), 1, 5, 769–773, 1979.

[LPS92] Julian Laderman, Victor Y. Pan, Xuan-He Sha, On Practical Algorithms for Accelerated Matrix Multiplication, Linear Algebra and Its Applications, 162–164, 557–588, 1992.

[MP80] Willard L. Miranker and Victor Y. Pan, Methods of Aggregations, Linear Algebra and Its Applications, 29, 231–257 (1980).

[MRT05] Per Gunnar Martinsson, Vladimir Rokhlin, Marc Tygert, A Fast Algorithm for the Inversion of Toeplitz Matrices, Computers and Mathematics with Applications, 50, 741–752, 2005.

[P66] Victor Y. Pan, On Methods of Computing the Values of Polynomials, Uspekhi Matematicheskikh Nauk, 21, 1(127), 103–134, 1966. [Transl. Russian Mathematical Surveys, 21, 1(127), 105–137, 1966.]

[P72] Victor Y. Pan, On Schemes for the Evaluation of Products and Inverses of Matrices (in Russian), Uspekhi Matematicheskikh Nauk, 27, 5 (167), 249–250, 1972.

[P78] Victor Y. Pan, Strassen's Algorithm Is Not Optimal: Trilinear Technique of Aggregating for Fast Matrix Multiplication, Proc. the 19th Annual IEEE Symposium on Foundations of Computer Science (FOCS'78), 166–176, IEEE Computer Society Press, Long Beach, California, 1978.

[P79] Victor Y. Pan, Fields Extension and Trilinear Aggregating, Uniting and Canceling for the Acceleration of Matrix Multiplication, Proceedings of the 20th Annual IEEE Symposium on Foundations of Computer Science (FOCS'79), 28–38, IEEE Computer Society Press, Long Beach, California, 1979.

[P80] Victor Y. Pan, New Fast Algorithms for Matrix Operations, SlAM J. on Computing, 9, 2, 321–342, 1980, and Research Report RC 7555, IBM T.J. Watson Research Center, February 1979.

[P81] Victor Y. Pan, New Combinations of Methods for the Acceleration of Matrix Multiplications, Computers and Mathematics (with Applications), 7, 1, 73–125, 1981.

[P82] Victor Y. Pan, Trilinear Aggregating with Implicit Canceling for a New Acceleration of Matrix Multiplication, Computers and Mathematics (with Applications), 8, 1, 23-34, 1982.

[P84a] Victor Y. Pan, How Can We Speed up Matrix Multiplication? SIAM Review, 26, 3, 393–415, 1984.

[P84b] Victor Y. Pan, How to Multiply Matrices Faster, Lecture Notes in Computer Science, 179, Springer, Berlin, 1984.

[P16] Victor Y. Pan, How Bad Are Vandermonde Matrices? SIAM Journal of Matrix Analysis and Applications, 37, 2, 676–694, 2016.

[P17] Victor Y. Pan, Fast Matrix Multiplication and Its Algebraic Neighborhood, SB MATH (Mathematical Sbornik), volume 208, issue 11, 2017 (in both Russian and English). DOI:10.1070/SM8833

[S69] Volker Strassen, Gaussian Elimination Is Not Optimal, Numerische Mathematik, 13, 354–356, 1969.

[S72] Volker Strassen, Evaluation of Rational Functions, in Analytical Complexity of Computations (edited by R.E. Miller, J. W. Thatcher, and J. D. Bonlinger), Plenum Press, New York (1972), pages 1–10.

[S74] Volker Strassen, Some Results in Algebraic Complexity Theory, Proceedings of the International Congress of Mathematicians, Vancouver, 1974 (Ralph D. James, editor), Volume 1, Canadian Mathematical Society (1974), pages 497–501.

[S81] Arnold Schönhage, Partial and Total Matrix Multiplication, SIAM J. on Computing, 10, 3, 434–455, 1981.

[W67] Shmuel Winograd, On the Number of Multiplications Required to Compute Certain Functions, Proc. National Academy of Sciences, 58, 1840–42, 1967.

[XXG12] Jianlin Xia, Yuanzhe Xi and Ming Gu, SIAM J. on Matrix Analysis and Applications, 33, 837–858, 2012.