

1970
Turing
Award
Lecture

Some Comments from a Numerical Analyst

J. H. WILKINSON
National Physical Laboratory
Teddington, Middlesex, England

A description is given of life with A. M. Turing at the National Physical Laboratory in the early days of the development of electronic computers (1946–1948). The present mood of pessimism among numerical analysts resulting from difficult relationships with computer scientists and mathematicians is discussed. It is suggested that in light of past and present performance this pessimism is unjustified and is the main enemy of progress in numerical mathematics. Some achievements in the fields of matrix computations and error analysis are discussed and likely changes in the direction of research in numerical analysis are sketched.

Introduction

When at last I recovered from the feeling of shocked elation at being invited to give the 1970 Turing Award Lecture, I became aware that I must indeed prepare an appropriate lecture. There appears to be a tradition that a Turing Lecturer should decide for himself what is expected from him, and probably for this reason previous lectures have differed considerably in style and content. However, it was made quite clear that I was to give an after-luncheon speech and that I would not have the benefit of an overhead projector or a blackboard.

Author deceased, October 1986. Address at that time: Department of Computer Science, Stanford University, Stanford, CA 94305.

Although I have been associated with high speed computers since the pioneering days, my main claim, such as it is, to the honor of giving the 1970 lecture rests on my work as a numerical analyst, particularly in the field of error analysis. A study of the program for this meeting revealed that numerical analysis was conspicuous by its absence, and accordingly I felt that it would be inappropriate to prepare a rather heavy discourse on rounding errors; indeed I doubt whether it would be a suitable topic for an after-luncheon speech in any setting. I decided therefore to make some rather personal comments based on my experience as a numerical analyst over the last twenty-five years.

There is one important respect in which it is reasonably probable that I shall occupy a unique position among Turing Lecturers.

Maurice Wilkes, giving the 1967 Turing Lecture remarked that it was unlikely that many of those who followed him would be people who were acquainted with Alan Turing. In fact I can claim a good deal more than that. From 1946 to 1948 I had the privilege of working with the great man himself at the National Physical Laboratory. I use the term "great man" advisedly because he was indeed a remarkable genius. To those of us at N.P.L. who knew him and worked with him, it has been a source of great pleasure that the ACM should have recognized his outstanding contributions to computer science by founding this Turing Award, and because of my connection with his work at an important period in his career, it is particularly gratifying for me to be a recipient. I trust that in the circumstances it will not be regarded as inappropriate if I devote a part of my lecture to the period I spent working with him. My career was certainly profoundly influenced by the association and, without it, it is unlikely that I would have remained in the computer field.

Life with Alan Turing

I was by inclination and training a classical analyst. Cambridge was still dominated by classical analysis in the '30s and I was strongly influenced by the Hardy-Littlewood tradition. Had it not been for World War II, I would almost certainly have taken my Ph.D. in that field. However, I was of military age when the war broke out and being by nature a patriotic man I felt that I could serve my country more effectively, and incidentally a lot more comfortably, working in the Government Scientific Service, than serving as an incompetent infantryman. The British Government took a surprisingly enlightened attitude on the subject, and almost from the start of the war those with scientific qualifications were encouraged to take this course.

I therefore spent the war in the Armament Research Department, which has much in common with Aberdeen Proving Ground, working mainly on such fascinating topics as external ballistics, fragmentation of bombs and shells, and the thermodynamics of explosives. My task was to solve problems of a mathematical nature arising in these fields,

using computational methods if necessary. Anybody who has ever been subjected to this discipline will know that it can be quite a chastening experience and successes are heavily diluted with failures. I did not at first find this task particularly congenial, but gradually I became interested in the numerical solution of physical problems. Later in this lecture I shall describe an early experience with matrix computations which was to have a considerable influence on my subsequent career.

It was not possible to obtain an immediate release at the end of the war and in 1946 I joined the newly formed Mathematics Division at the National Physical Laboratory. It was there that I first met Alan Turing, though he was, of course, known to me before by reputation, but mainly as an eccentric. It is interesting to recall now that computer science virtually embraces two whole divisions at N.P.L. and spreads its tentacles into the remainder, that at that time the staff of the high speed computing section (or ACE section as it was called) numbered $1\frac{1}{2}$. The one, of course, was no less a person than Alan Turing himself and I was the half. I hasten to add that this doesn't represent false modesty on my part. I was to spend half my time in the Computing Section, which was in the capable hands of Charles Goodwin and Leslie Fox, and the other half with Alan Turing. For several months Alan and I worked together in a remarkably small room at the top of an old house which had been taken over "temporarily" by N.P.L. to house Mathematics Division. Needless to say, twenty-five years later it is still part of N.P.L. Turing never became an empire builder; he assembled his staff rather slowly and worked rather intimately with them. A year later the staff had reached only $3\frac{1}{2}$, the two additions being Mike Woodger, who is best known for his work on Algol, and Harry Huskey (who needs no introduction to ACM audiences) who spent 1947 at N.P.L.

My task was to assist Turing in the logical design of the computer ACE which was to be built at N.P.L. and to consider the problems of programming some of the more basic algorithms of numerical analysis, and my work in the Computing Section was intended to broaden my knowledge of that subject. (Those of you who are familiar with Turing's work will be interested to know that he referred to the sets of instructions needed for a particular problem as the relevant "instruction table," a term which later led to misunderstandings with people elsewhere.) As you can imagine, this left me with little idle time. Working with Turing was tremendously stimulating, perhaps at times to the point of exhaustion. He had recently become keenly interested in the problems of numerical analysis himself, and he took great pleasure in subjecting Leslie Fox, who was our most experienced numerical analyst at N.P.L., to penetrating but helpful criticisms of the methods he was using.

It was impossible to work "half-time" for a man like Turing and almost from the start the periods spent with the Computing Section were rather brief. The joint appointment did, however, have its useful

aspect. Turing occasionally had days when he was "unapproachable" and at such times it was advisable to exercise discretion. I soon learned to recognize the symptoms and would exercise my right (or, as I usually put it, "meet my obligations") of working in the Computing Section until the mood passed, which it usually did quite quickly.

Turing had a strong predilection for working things out from first principles, usually in the first instance without consulting any previous work on the subject, and no doubt it was this habit which gave his work that characteristically original flavor. I was reminded of a remark which Beethoven is reputed to have made when he was asked if he had heard a certain work of Mozart which was attracting much attention. He replied that he had not, and added "neither shall I do so, lest I forfeit some of my own originality."

Turing carried this to extreme lengths and I must confess that at first I found it rather irritating. He would set me a piece of work and when I had completed it he would not deign to look at my solution but would embark on the problem himself; only after having a preliminary trial on his own was he prepared to read my work. I soon came to see the advantage of his approach. In the first place he was really not as quick at grasping other people's ideas as he was at formulating his own, but what is more important, he would frequently come up with some original approach which had escaped me and might well have eluded him, had he read my account immediately. When he finally got around to reading my own work he was generally very appreciative; he was particularly fond of little programming tricks (some people would say that he was too fond of them to be a "good" programmer) and would chuckle with boyish good humor at any little tricks I may have used.

When I joined N.P.L. I had not made up my mind to stay permanently and still thought in terms of returning to Cambridge to take up research in classical analysis. The period with Turing fired me with so much enthusiasm for the computer project and so heightened my interest in numerical analysis that gradually I abandoned this idea. As I rather like to put it when speaking to pure mathematical friends, "had it not been for Turing I would probably have become just a pure mathematician," taking care to give the remark a suitably pejorative flavor.

Turing's reputation is now so well established that it scarcely stands in need of a boost from me. However, I feel bound to say that his published work fails to give an adequate impression of his remarkable versatility as a mathematician. His knowledge ranged widely over the whole field of pure and applied mathematics and seemed, as it were, not merely something he had learned from books, but to form an integral part of the man himself. One could scarcely imagine that he would ever "forget" any of it. In spite of this he had only twenty published papers to his credit (and this only if one includes virtually everything), written over a period of some twenty years. Remarkable

as some of these papers are, this work represents a mere fraction of what he might have done if things had turned out just a little differently.

In the first place there were the six years starting from 1939 which he spent at the Foreign Office. He was 27 in 1939, so that in different circumstances this period might well have been the most productive of his life. He seemed not to have regretted the years he spent there and indeed we formed the impression that this was one of the happiest times of his life. Turing simply loved problems and puzzles of all kinds and the problems he encountered there must have given him a good deal of fun. Certainly it was there that he gained his knowledge of electronics and this was probably the decisive factor in his deciding to go to N.P.L. to design an electronic computer rather than returning to Cambridge. Mathematicians are inclined to refer to this period as the "wasted years" but I think he was too broad a scientist to think of it in such terms.

A second factor limiting his output was a marked disinclination to put pen to paper. At school he is reputed to have had little enthusiasm for the "English subjects" and he seemed to find the tedium of publishing a paper even more oppressive than most of us do. For myself I find his style of writing rather refreshing and full of little personal touches which are particularly attractive to someone who knew him. When in the throes of composition he would hammer away on an old typewriter (he was an indifferent typist, to put it charitably) and it was on such occasions that visits to the Computing Section were particularly advisable.

While I was preparing this talk, an early Mathematics Division report was unearthed. It was written by Turing in 1946 for the Executive Committee of N.P.L., and its main purpose was to convince the committee of the feasibility and importance of building an electronic computer. It is full of characteristic touches of humor, and rereading it for the first time for perhaps 24 years I was struck once again by his remarkable originality and versatility. It is perhaps salutary to be reminded that as early as 1946 Turing had considered the possibility of working with both interval and significant digit arithmetic and the report recalled forgotten conversations, not to mention heated arguments, which we had on this topic.

Turing's international reputation rests mainly on his work on computable numbers but I like to recall that he was a considerable numerical analyst, and a good part of his time from 1946 onwards was spent working in this field, though mainly in connection with the solution of physical problems. While at N.P.L. he wrote a remarkable paper on the error analysis of matrix computations [1] and I shall return to this later.

During the last few months at N.P.L., Turing became increasingly dissatisfied with progress on the ACE project. He had always thought in terms of a large machine with 200 long relay lines storing some 6,000

words and I think this was too ambitious a project for the resources of N.P.L. (and indeed of most other places) at that time. During his visit Harry Huskey attempted to get work started on a less ambitious machine, based on Turing's ideas. Alan could never bring himself to support this project and in 1948 he left N.P.L. to join the group at Manchester University. After he left, the four senior members of the ACE section of Mathematics Division and the recently formed Electronics Section joined forces and collaborated on the construction of the computer PILOT ACE, for which we took over some of the ideas we had worked out with Harry Huskey; for the next two to three years we all worked as electronic engineers. I think we can claim that the PILOT ACE was a complete success and since Turing would not have permitted this project to get off the ground, to this extent at least we benefitted from his departure, though the Mathematics Division was never quite the same again. Working with a genius has both advantages and disadvantages! Once the machine was a success, however, there were no sour grapes from Turing and he was always extremely generous about what had been achieved.

The Present State of Numerical Analysis

I would now like to come to the main theme of my lecture, the present status of numerical analysis. Numerical analysis is unique among the various topics which comprise the rather ill-defined discipline of computer science. I make this remark rather defiantly because I would be very sorry to see numerical analysis sever all its connections with computer science, though I recognize that my views must be influenced to some extent by having worked in the exciting pioneer days on the construction of electronic computers. Nevertheless, numerical analysis is clearly different from the other topics in having had a long and distinguished history. Only the name is new (it appears not to have been used before the '50s) and this at least it has in common with computer science.

Some like to trace its history back to the Babylonians and if one chooses to regard any reasonable systematic computation as numerical analysis I suppose this is justifiable. Certainly many of the giants of the mathematical world, including both the great Newton and Gauss themselves, devoted a substantial part of their research to computational problems. In those days it was impossible for a mathematician to spend his time in this way without being apprehensive of the criticism of his colleagues.

Many of the leaders of the computer revolution thought in terms of developing a tool which was specifically intended for the solution of problems arising in physics and engineering. This was certainly true of the two men of genius, von Neumann and Turing, who did so much to attract people of real ability into the computing field in the early

days. The report of Turing to which I referred earlier makes it quite clear that he regarded such applications as the main justification for embarking on what was, even then, a comparatively expensive undertaking. A high percentage of the leading lights of the newly formed computer societies were primarily numerical analysts and the editorial boards of the new journals were drawn largely from their ranks.

The use of electronic computers brought with it a new crop of problems all perhaps loosely associated with "programming" and quite soon a whole field of new endeavors grew up around the computer. In a brilliant article on numerical analysis [2] Philip Davis uses the term "computerology" to encompass these multifarious activities but is careful to attribute the term to an unnamed friendly critic. I do not intend to use the term in a pejorative sense in this talk, but it is a useful collective word to cover everything in computer science other than numerical analysis. Many people who set out originally to solve some problem in mathematical physics found themselves temporarily deflected by the problems of computerology and we are still waiting with bated breath for the epoch-making contributions they will surely make when they return to the fold, clothed in their superior wisdom.

In contrast to numerical analysis the problems of computerology are entirely new. The whole science is characterized by restless activity and excitement and completely new topics are constantly springing up. Although, no doubt, a number of the new activities will prove to be short-lived, computerology has a vital part to play in ensuring that computers are fully exploited. I'm sure that it is good for numerical analysts to be associated with a group of people who are so much alive and full of enthusiasm. I'm equally sure that there is merit in computer science embracing a subject like numerical analysis which has a solid background of past achievement. Inevitably though, numerical analysis has begun to look a little square in the computer science setting, and numerical analysts are beginning to show signs of losing faith in themselves. Their sense of isolation is accentuated by the present trend towards abstraction in mathematics departments which makes for an uneasy relationship. How different things might have been if the computer revolution had taken place in the 19th century! In his article Davis remarks that people are already beginning to ask, "Is numerical analysis dead?" Davis has given his own answer to this question and I do not propose to pursue it here. In any case "numerical analysts" may be likened to "The Establishment" in computer science and in all spheres it is fashionable to diagnose "rigor mortis" in the Establishment.

There is a second question which is asked with increasing frequency. It assumes many different guises but is perhaps best expressed by the catch-phrase, "What's new in numerical analysis?" This is invariably delivered in such a manner as to leave no doubt that the questioner's answer is "Nothing," or, even more probably, one of the more vulgar two-word synonyms, in which the English language is so rich. This

criticism reminds me of a somewhat similar situation which exists with respect to functional analysis. Those brought up in an older tradition are inclined to say that "there is nothing new in functional analysis, it merely represents a dressing up of old results in new clothes." There is just sufficient truth in this to confirm the critics in their folly.

In my opinion the implied criticism involves a false comparison. Of course everything in computerology is new; that is at once its attraction, and its weakness. Only recently I learned that computers are revolutionizing astrology. Horoscopes by computer! — it's certainly never been done before, and I understand that it is very remunerative! Seriously though, it was not to be expected that numerical analysis would be turned upside down in the course of a decade or two, just because we had given it a new name and at last had satisfactory tools to work with. Over the last 300 years some of the finest intellects in the mathematical world have been brought to bear on the problems we are trying to solve. It is not surprising that our rate of progress cannot quite match the heady pace which is so characteristic of computerology.

Some Achievements in Numerical Analysis

In case you are tempted to think that I am about to embark on excuses for not having made any real progress, I hasten to assure you that I have no such intention. While I was preparing this lecture I made a brief review of what has been achieved since 1950 and found it surprisingly impressive. In the next few minutes I would like to say just a little about the achievements in the area with which I am best acquainted, matrix computations.

We are fortunate here in having, in the little book written by V. N. Faddeeva [3], an admirably concise and accurate account of the situation as it was in 1950. A substantial part of the book is devoted to the solution of the eigenvalue problem, and scarcely any of the methods discussed there are in use today. In fact as far as non-Hermitian matrices are concerned, even the methods which were advocated at the 1957 Wayne matrix conference have been almost completely superseded. Using a modern version of the QR algorithm one can expect to produce an accurate eigensystem of a dense matrix of order 100 in a time which is of the order of a minute. One can then go on to produce rigorous error bounds for both the eigenvalues and eigenvectors if required, deriving a more accurate system as a by-product. At the 1957 Wayne conference we did not appear to be within hailing distance of such an achievement. A particularly pleasing feature of the progress is that it is an appreciation of the problem of numerical stability resulting from advances in error analysis that has played a valuable part in suggesting the new algorithms.

Comparable advances have been made in the development of iterative methods for solving sparse linear systems of the type arising from the partial differential equations; here algorithmic advances have proceeded *pari passu* with a deepening understanding of the convergence properties of iterative methods. As far as dense systems are concerned the development of new algorithms has been less important, but our understanding of the stability of the standard methods has undergone a complete transformation.

In this connection I would like to make my last remarks about life with Turing. When I joined N.P.L. in 1946 the mood of pessimism about the stability of elimination methods for solving linear systems was at its height and was a major talking point. Bounds had been produced which purported to show that the error in the solution would be proportional to 4^n and this suggested that it would be impractical to solve systems even of quite modest order. I think it was true to say that at that time (1946) it was the more distinguished mathematicians who were most pessimistic, the less gifted being perhaps unable to appreciate the full severity of the difficulties. I do not intend to indicate my place on this scale, but I did find myself in a rather uncomfortable position for the following reason.

It so happens that while I was at the Armament Research Department I had an encounter with matrix computations which puzzled me a good deal. After a succession of failures I had been presented with a system of twelve linear equations to solve. I was delighted at last at being given a problem which I "knew all about" and had departed with my task, confident that I would return with the solution in a very short time. However, when I returned to my room my confidence rapidly evaporated. The set of 144 coefficients suddenly looked very much larger than they had seemed when I was given them. I consulted the few books that were then available, one of which, incidentally, recommended the direct application of Cramer's rule using determinants! It did not take long to appreciate that this was not a good idea and I finally decided to use Gaussian elimination with what would now be called "partial pivoting."

Anxiety about rounding errors in elimination methods had not yet reared its head and I used ten-decimal computation more as a safety precaution rather than because I was expecting any severe instability problems. The system was mildly ill-conditioned, though we were not so free with such terms of abuse in those days, and starting from coefficients of order unity, I slowly lost figures until the final reduced equation was of the form, say,

$$.0000376235x_{12} = .0000216312$$

At this stage I can remember thinking to myself that the computed x_{12} derived from this relation could scarcely have more than six correct

figures, even supposing that there had been no buildup in rounding errors, and I contemplated computing the answers to six figures only. However, as those of you who have worked with a desk computer will know, one tends to make fewer blunders if one adheres to a steady pattern of work, and accordingly I computed all variables to ten figures, though fully aware of the absurdity of doing so. It so happened that all solutions were of order unity, which from the nature of the physical problem was to be expected.

Then, being by that time a well-trained computer, I substituted my solution in the original equations to see how they checked. Since x_1 had been derived from the first of the original equations, I started by substituting in the 12th equation. You will appreciate that on a desk machine the inner-product is accumulated exactly giving 20 figures in all. (It is interesting that nowadays we nearly always accept a poorer performance from the arithmetic units of computers!) To my astonishment the left-hand side agreed with the given right-hand side to ten figures, i.e., to the full extent of the right-hand side. That, I said to myself, was a coincidence. Eleven more "coincidences" followed, though perhaps not quite in rapid succession! I was completely baffled by this. I felt sure that none of the variables could have more than six correct figures and yet the agreement was as good as it would have been if I had been given the exact answer and had then rounded it to ten figures. However, the war had still to be won, and it was no time to become introspective about rounding errors; in any case I had already taken several times longer than my first confident estimate. My taskmaster was not as appreciative as he might have been but he had to admit he was impressed when I claimed that I had "the exact solution" corresponding to a right-hand side which differed only in the tenth figure from the given one.

As you can imagine this experience was very much in my mind when I arrived at N.P.L. and encountered the preoccupation with the instability of elimination methods. Of course I still believed that my computed answers had at best six correct figures, but it was puzzling that in my only encounter with linear systems it was the surprising *accuracy* of the solutions (at least in the sense of small residuals) which required an explanation. In the current climate at N.P.L. I decided not to risk looking foolish by stressing this experience.

However, it happened that some time after my arrival, a system of 18 equations arrived in Mathematics Division and after talking around it for sometime we finally decided to abandon theorizing and to solve it. A system of 18 is surprisingly formidable, even when one had previous experience with 12, and we accordingly decided on a joint effort. The operation was manned by Fox, Goodwin, Turing, and me, and we decided on Gaussian elimination with complete pivoting. Turing was not particularly enthusiastic, partly because he was not an experienced performer on a desk machine and partly because he was convinced that it would be a failure. History repeated itself remarkably

closely. Again the system was mildly ill-conditioned; the last equation had a coefficient of order 10^{-4} (the original coefficients being of order unity) and the residuals were again of order 10^{-10} , that is, of the size corresponding to the exact solution rounded to ten decimals. It is interesting that in connection with this example we subsequently performed one or two steps of what would now be called "iterative refinement," and this convinced us that the first solution had had almost six correct figures.

I suppose this must be regarded as a defeat for Turing since he, at that time, was a keener adherent than any of the rest of us to the pessimistic school. However, I'm sure that this experience made quite an impression on him and set him thinking afresh on the problem of rounding errors in elimination processes. About a year later he produced his famous paper "Rounding-off errors in matrix processes" [1] which together with the paper of J. von Neumann and H. Goldstine [4] did a great deal to dispel the gloom. The second round undoubtedly went to Turing!

This anecdote illustrates rather well, I think, the confused state of mind which existed at that time, and was shared even by the most distinguished people working in the field. By contrast I think we can fairly claim today to have a reasonably complete understanding of matrix stability problems, not only in the solution of linear systems, but also in the far more difficult eigenvalue problem.

Failures in the Matrix Field

Although we can claim to have been successful in the matrix area as far as the development of algorithms and an understanding of their performance is concerned, there are other respects in which we have not been particularly successful even in this field. Most important of these is a partial failure in communication. The use of algorithms and a general understanding of the stability problem has lagged much further behind developments than it should have. The basic problems of matrix computation have the advantage of simple formulations, and I feel that the preparation of well-tested and well-documented algorithms should have advanced side by side with their development and analysis. There are two reasons why this has not happened. (i) It is a much more arduous task than was appreciated to prepare the documentation thoroughly. (ii) Insufficient priority has been attached to doing it. There are signs in the last year or two that these shortcomings are at last being overcome with the work on the *Handbook for Automatic Computation* [5], that on matrix algorithms centered at Argonne National Laboratory, and the more general project at Bell Telephone Laboratories [6]. I think it is of vital importance that all the work that has been expended on the development of satisfactory algorithms should be made fully available to the people who need to use it. I would go further than this and claim that it is a social duty to see that this is achieved.

A second disquieting feature about work in the matrix field is that it has tended to be isolated from that in very closely related areas. I would like to mention in particular linear programming and statistical computations. Workers in linear algebra and linear programming seemed until recently to comprise almost completely disjoint sets and this is surely undesirable. The standard computations required in practical statistics provide the most direct opportunities for applying the basic matrix algorithms and yet there is surprisingly little collaboration. Only recently I saw an article by a well-known practical statistician on the singular value decomposition which did not, at least in its first draft, contain any reference to the work of Kahan and Golub who have developed such an admirable algorithm for this purpose. Clearly there is a failure on both sides, but I think it is primarily the duty of people working in the matrix field to make certain that their work is used in related areas, and this calls for an aggressive policy. Again there are signs that this isolation is breaking down. At Stanford, Professor Dantzig, a pioneer in linear programming, now has a joint appointment with the Computer Science Department and schemes are afoot in the UK to have joint meetings of matrix experts and members of the Statistical Society. Historical accidents often play a great part in breaking down barriers and it is interesting that collaboration between workers on the numerical solution of partial differential equations and on matrix algebra has always been extremely close.

A third disappointing feature is the failure of numerical analysts to influence computer hardware and software in the way that they should. In the early days of the computer revolution computer designers and numerical analysts worked closely together and indeed were often the same people. Now there is a regrettable tendency for numerical analysts to opt out of any responsibility for the design of the arithmetic facilities and a failure to influence the more basic features of software. It is often said that the use of computers for scientific work represents a small part of the market and numerical analysts have resigned themselves to accepting facilities "designed" for other purposes and making the best of them. I am not convinced that this is inevitable, and if there were sufficient unity in expressing their demands there is no reason why they could not be met. After all, one of the main virtues of an electronic computer from the point of view of the numerical analyst is its ability to "do arithmetic fast." Need the arithmetic be so bad! Even here there are hopeful developments. The work of W. Kahan deserves particular mention and last September a well-known manufacturer sponsored a meeting on this topic at which he, among others, had an opportunity to express his views.

Final Comments

I am convinced that mathematical computation has a great part to play in the future and that its contribution will fully live up to the expectations of the great pioneers of the computer revolution. The

greatest danger to numerical analysts at the moment springs from a lack of faith in themselves for which there is no real justification. I think the nature of research in numerical analysis is bound to change substantially in the next decade. In the first two decades we have concentrated on the basic problems, such as arise, for example, in linear and nonlinear algebra and approximation theory. In retrospect these will appear as a preliminary sharpening of the tools which we require for the real task. For success in this it will be essential to recruit more effectively than we have so far from the ranks of applied mathematicians and mathematical physicists. On a recent visit to the Soviet Union I was struck by the fact that most of the research in numerical analysis is being done by people who were essentially mathematical physicists, who have decided to tackle their problems by numerical methods, and they are strongly represented in the Academy of Sciences. Although I think that we in the West have nothing to fear from a comparison of achievements, I do feel that morale is markedly higher in the Soviet Union.

In the UK there are signs that the tide is already turning. There is to be a Numerical Analysis Year at the University of Dundee, during the course of which many of the more distinguished of the world's numerical analysts will be visiting the UK. Quite recently a Discussion Meeting on a numerical analysis topic was held at the Royal Society. Such things would scarcely have been contemplated a year or two ago. I look forward to the time when numerical mathematics will dominate the applied field and will again occupy a central position at meetings of the ACM.

References

1. Turing, A. M. Rounding-off errors in matrix processes. *Quart. J. Mech.* 1 (1948), 287-308.
2. Davis, P. J. Numerical analysis. In *The Mathematical Sciences: A Collection of Essays*. MIT Press, Cambridge, Mass., 1969.
3. Faddeeva, V. N. *Computational Methods of Linear Algebra*, Translated by C. D. Benster. Dover, New York, 1959.
4. von Neumann, J., and Goldstine, H. H. Numerical inverting of matrices of high order. *Bull. Amer. Math. Soc.* 53 (1947), 1021-1099.
5. Wilkinson, J. H. *Handbook for Automatic Computation, Vol. 2. Linear Algebra*. Springer-Verlag, Berlin.
6. Gentleman, W. M., and Traub, J. F. The Bell Laboratories numerical mathematics program library project. Proc. ACM 23rd Nat. Conf., 1968, Brandon/Systems Press, Princeton, N.J., pp. 485-490.

Categories and Subject Descriptors:

F.2.1 [Theory of Computation]: Numerical Algorithms and Problems—*Computations on Matrices*; G.1.0 [Mathematics of Computing]: Numerical Analysis—*error analysis; numerical algorithms*; G.1.3 [Mathematics of Computing]: Numerical Linear Algebra—*linear systems*; K.2 [Computing Milieux]: History of Computing—*people*

General Terms:

Algorithms, Theory

Additional Keywords and Phrases:

National Physical Laboratory, Turing