

N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM
MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT
CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED
IN THE INTEREST OF MAKING AVAILABLE AS MUCH
INFORMATION AS POSSIBLE

JPL PUBLICATION 79-106

Symbolic-Numeric Interface: A Review

Edward W. Ng

(NASA-CR-162690) SYMBOLIC-NUMERIC
INTERFACE: A REVIEW (Jet Propulsion Lab.)
21 p HC A02/MF A01 CSCL 09B

N80-16768

Unclas
G3/61 46974

January 1, 1980

National Aeronautics and
Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California



JPL PUBLICATION 79-106

Symbolic-Numeric Interface: A Review

Edward W. Ng

January 1, 1980

National Aeronautics and
Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

The research described in this publication was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under NASA Contract No. NAS7-100 and under NSF Grant No. MCS76-22939.

CONTENTS

I.	Prologue	1
II.	From the Monologues of Two Leading Scholars	2
III.	A Decade of Dialogues	3
IV.	A Catalogue of Activities	4
V.	My Decalogue of Favorite Applications	6
VI.	The Power and Limitation of Analogues	11
VII.	Epilogue	12
VIII.	A Log of Acknowledgements	13
	References	14

PRECEDING PAGE BLANK NOT FILMED

PREFACE

The material in this publication was originally presented as a paper at EUROSAM '79, An International Symposium on Symbolic and Algebraic Manipulation, Marseille, France, June 1979, and appeared in the proceedings published by Springer-Verlag.

ABSTRACT

This is a survey of recent activities that either used or encouraged the potential use of a combination of symbolic and numerical calculations. Symbolic calculations here primarily refer to the computer processing of procedures from classical algebra, analysis and calculus. Numerical calculations refer to both numerical mathematics research and scientific computation. This survey is intended to point out a large number of problem areas where a co-operation of symbolic and numeric methods is likely to bear many fruits. These areas include such classical operations as differentiation and integration, such diverse activities as function approximations and qualitative analysis, and such contemporary topics as finite element calculations and computational complexity. It is contended that other less obvious topics such as the Fast Fourier Transform, linear algebra, nonlinear analysis and error analysis would also benefit from a synergistic approach advocated here.

SYMBOLIC-NUMERIC INTERFACE: A REVIEW

I. Prologue

As we approach the end of a decade, it is appropriate to review the progress (or lack of it) of various research activities. One topic being appropriate for such review concerns the interface between symbolic and numerical computation. It is probably safe to conjecture that the majority of scientific and engineering computation involves a combination of arithmetic, algebraic and analytic investigations. It is a common practice to exploit the computer to perform the arithmetic investigations, leaving the a priori or a posteriori analytic or algebraic investigations to the human, by manual operations. Though we have witnessed in this decade increasing efforts to exploit the computer for these analytic investigations, it is still somewhat disconcerting to see a relatively small amount of computerized arithmetic-analytic investigations.

In order to better focus on the interface between symbolic and numerical computations, we need to first characterize the two spheres of activities in question. By numerical computation, I refer to two broad classes of activities, viz., numerical mathematics research and development, and numerical calculations in

science and engineering. The former class is concerned with tool construction while the latter with mathematical modeling or simulation. To further refine this characterization, we may cite Traub's (1972) description of the four major components of numerical mathematics research: (1) foundations, (2) synthesis and analysis of algorithms, (3) analysis of error, (4) programs and program libraries. The other class of numerical calculations, alias scientific computation, does not possess such a convenient compact description, and in fact is characterized by a diffusion of ideas and methods, as best summarized by Rice (1972): "Scientific computation is going everywhere science is going; scientific computation, independent of the science involved, will evolve significantly."

Turning to symbolic computation, we shall use as a frame of reference Sammet's trilogy (1971): (A) theorem proving, (B) pure mathematics, and (C) formula manipulation. Here (A) essentially involves the computer simulation of procedures in mathematical logic. (B) primarily indicates attempts to apply computers to modern algebra, number theory, combinatorics, graph theory and other more abstract topics. (C) is concerned with computer processing of procedures from classical algebra, analysis and calculus. It is this last item that generates the most interface with numeric computation, and that will occupy the bulk of this presentation.

II. From the Monologues of Two Leading Scholars

In the consideration of symbolic-numeric interface, we believe that two epistemological issues stand out, namely, approximation and man-computer symbiosis. At the end of the last decade, Birkhoff (1969), in his Retiring Presidential Address for SIAM, gave an interesting overview of the interface between mathematics and psychology.

Throughout his address, Birkhoff emphasized the importance of approximation, which served as a bridge between the discrete and the continuum. He cited psychological, sensory and epistemological reasons for the stature of approximations in our intellectual enterprises. He first argued that the human mind was a least a hybrid computer, with capability for both digital (discrete) and analogue (continuous) functions. He then cited our empirical senses as satisfied with approximations of continuous space-time (e.g. motion picture), and gave psychological evidence that "we humans have two qualitatively different modes of mathematical thinking, which are rooted in our behavioral techniques for counting and measuring, and are inherent in our physiological makeup" (p. 446, top paragraph). To further honour the epistemological stature of approximations, Birkhoff attributed the impressive success of computing in physics and engineering to "the skillful use of approximations, whose details are understood only by skilled human numerical analyst" (p. 444, 3rd paragraph). In our opinion, the need and power of approximations will motivate much activity in the symbiosis of symbolic and numeric methods.

His further excursion into the psychology of mathematics led Birkhoff to

the consideration of man-computer symbiosis. Earlier in the decade Licklider (1960) argued prophetically for the case of such symbiosis. He cited the main aims of such co-operation as (1) to let computers facilitate formulative thinking as they facilitated the solution of formulated problems, and (2) to enable men and computers to cooperate in making decisions and controlling complex situations without inflexible dependence on predetermined programs. We believe that symbolic-numeric interface is most sensitive to, and yet constrained by, but also enhanced by the mode of man-computer symbiosis.

III. A Decade of Dialogues

When we consider dialogues that affect symbolic-numeric interfaces, we may think of the symbolic mathematician's dialogues with the numerical mathematician, the scientist-engineer user, and the systems designer-programmer. First, we turn our attention to more collective dialogues in this decade. In this section we shall only attend to the interface in form, leaving to the next three sections to describe the interface in substance.

We first attend to one front of communications, namely, between symbolic and numerical mathematicians. In the beginning of this decade, John Rice was instrumental in starting the "mathematical software movement". His concept of mathematical software had always included numerical, symbolic, and statistical software. Thus in his first symposium on mathematical software, April 1970 at Purdue, Jean Sammet was invited to lecture on "Software for Nonnumerical Mathematics" (Rice 1971). A year later, in the 2nd Symposium on Symbolic and Algebraic Manipulation (SYMSAM II, Mar. 1971), Sammet referred to that experience as "Daniel in the lion's den". As an observer in the Purdue Symposium, I can testify that Dr. Sammet was actually "Danielle in a den of indifferent lions". But that was the genesis of the dialogue, even though Sammet's paper took up only about 5% of the Symposium time and proceedings pages. In SYMSAM II we saw the table turned. There one leading numerical mathematician, Cleve Moler, braved the symbolic world with his paper on semi-symbolic methods in PDE. But Moler was not as lonely as Sammet in the Purdue occasion. His was at least flanked by two other papers exploiting the symbolic-numeric border, by Oman and Chepurnig (cf. SYMSAM II Proceedings, 1971). Since those encounters in the early seventies, things have improved considerably. Under the auspices of SIAM and ACM, there were more activities encouraging the symbolic-numeric symbiosis. We may cite the three mini-symposia in national SIAM meetings (Hampton, Ca. 1978; Albuquerque, N.M. 1977; Madison, Wis. 1978). We may cite the cooperation in the ACM Transactions on Mathematical Software. We may cite positive advocations by leaders in both camps. For example, the statement by Traub in a SIGNUM panel held in IFIP Congress 71: "I believe that a mixed symbolic numeric approach is powerful." (Traub, 1972) This sentiment was recently echoed by Hearn and Brown in their participation in

a COSERS study: "We conclude that the fields of symbolic and numerical computation can advance most fruitfully in harmony rather than in competition" (Brown and Hearn, 1979). However, despite all these positive signs, the interface between the two camps of researchers has contributed more to form than to substance. It is our hope that the cooperation in form will generate more activities in substance in the next decade.

Turning to another front of dialogues, we witness more substantive encounters between the symbolic mathematician and his scientific and engineering user. If we look at the major symposia sponsored by SIGSAM in this decade, we may get a pessimistic impression that "all is quiet on this front". For example, we saw only a handful of papers on scientific computing in SYMSAM II, EUROSAM 74 and SYMSAC 76 (cf. Proceedings of these symposia). But these symposia are poor indicators of the intensity of dialogues between symbolic mathematicians and their scientific users. After all, how many of us who use operating systems and compilers would go to symposia on those topics.

On the unstructured front we see impressive success by Hearn whose REDUCE has penetrated and made an impact in the physics world. He recently initiated a REDUCE newsletter which encouraged informal dialogues in the REDUCE community. The first issue listed a bibliography of 123 REDUCE references which include a large portion of papers of scientific applications of REDUCE (Hearn, 1978). He is the one symbolic mathematician most invited to lecture to the external scientific community. Hearn's success added credence to the thesis that unstructured exchange is still a most effective form of scientific dialogue.

Turning to more structured forums, we may cite the sessions in SIAM and ACM national meetings. For the type of cross-fertilization being considered here, we believe SIAM meetings represent far more fruitful grounds. In addition to these, recent attempts of user's conferences, tutorial seminars, and meetings on computational physics, are all healthy steps towards such dialogues. We may cite as examples the MACSYMA User's Conference (MACSYMA 1977), the IAC/CNR Tutorial Seminar (SIGSAM 1978, p. 10), and the International Colloquium on Advanced Computing Methods in Theoretical Physics (SIGSAM 1978, p. 7).

Having looked at the various avenues and forums of dialogue, we now turn to the substance of these interchanges.

IV. A Catalogue of Activities

Recently Brown and Hearn (1978) wrote a survey on the applications of symbolic algebraic computation. They concluded that most applications also involved numerical computation in some way. Therefore, they emphasized the interfaces between the two types of computation and advocated more cooperation. In many ways the present paper is supplementary to their survey. Therefore, we first summarize the Brown-Hearn paper, treating it as a catalogue of such symbolic applications. The authors enumerated the following topics.

(1) Popular application areas

Two areas are identified as most popular for the applications of symbolic computation, viz., celestial mechanics and quantum electro-dynamics. Both areas of applications involve perturbation theories, and in both cases a symbolic-numeric symbiosis is necessary to obtain meaningful results. A third area of application may be loosely grouped under the heading of general relativity. This area involves the processing of more general and complex expressions and therefore is much more sensitive to the issue of algebraic simplification. Here the symbolic-numeric interface is less important.

(2) Numerical evaluation of symbolic results

This topic may best be summarized in the words of Brown and Hearn: "When a problem in applied mathematics can be solved completely in symbolic form, the results may provide both qualitative insight and a basis for quantitative predictions. To facilitate such predictions, most symbolic algebra systems permit the replacement of the indeterminates and parameters in a symbolic expression by floating-point numbers. However, these general substitution mechanisms tend to be quite inefficient, and it is best to generate a special numerical evaluation program whenever a symbolic expression must be evaluated many times." In the context of symbolic-numeric interface this type of application is most susceptible to abuses. Therefore we shall render further discussion of this topic in the next section.

(3) Hybrid problems

This name is used by Hearn and Brown to describe problems that require a combination of symbolic and numerical techniques for their solution. As mentioned before, we suspect most scientific and engineering problems require a combination of analytic and numerical techniques, but only a small percentage takes advantage of symbolic tools in the analytic phase.

The two popular areas mentioned above, celestial mechanics and quantum electro-dynamics, are certainly rich with examples of such hybrid problems. Other such problems may be found in functional approximations, error analysis and multidimensional integrations. More detailed examples will be given later.

(4) Hybrid methods

As distinguished from the previous sub-topic, this one considers general methods for the solution of a class of problems. In the Brown and Hearn paper, three examples are mentioned of general methods for classes of hybrid problems, viz., finite element computation, Taylor expansion of ordinary differential equations, and numerical solution of nonlinear equations. For finite element computation, the symbolic portion involves the integration of shape functions. For the Taylor application, it involves the processing of some recurrence relations, which in turn requires manipulation of transcendental and algebraic functions. In the case of the nonlinear equation solving, the

most needed symbolic tool is multivariate differentiation with some built-in simplification. We shall further explain this item in the next section.

(5) Hybrid data

Usually, symbolic computation involves mathematical expressions of unpredictable size and shape in which the coefficients are exact integers and rational numbers, while numerical computation involves fixed precision floating-point numbers. Brown and Hearn use this phrase to describe computation that involves data with attributes of both these customary forms. Examples in this realm of computation include (a) symbolic expressions with floating-point coefficients, (b) floating-point numbers of arbitrary (dynamically determined) precision, (c) floating-point intervals of arbitrary precision and (d) exact rational intervals. In some computational contexts, these hybrid representations may serve the useful purpose of generating results of guaranteed accuracy. However, extreme care must be exerted to insure that such representations are relevant. For example, one reason why interval arithmetic has not been popular in applications is due to the impractical bound one obtains after a sequence of arithmetic operations.

In summary, Brown and Hearn gave a rather detailed survey of the two way traffic between numerical and symbolic computations. However, as they indicated, "as there are now over 500 papers which consider some aspect or application of symbolic computation, we could not hope to present a complete review of the field."

V. My Decalogue of Favorite Applications

The word 'decalogue' is used here in an unorthodox sense, to mean 'ten items' instead of 'ten commandments'. The intention here is to highlight some of the problems and prospects in the interfaces of numeric and symbolic computation, and the emphasis is more in the illustrative than the exhaustive. Many of the examples are taken from the survey by Brown and Hearn.

(1) Numerical evaluation of symbolic results

Tobey (1971) identified three basic approaches to such symbolic-numeric conversion capability: (a) interpretive evaluation, (b) direct code generation, and (c) generation of arithmetic statements of a programming language. There is actually a variety of approaches that are variances to one of Tobey's categories. They shall be referred to subsequently. As mentioned before, interpretive evaluation tends to be rather inefficient and therefore should be used only for 'one shot' application. Tobey's alternative (b) was popular for earlier symbolic systems. For example, the older FORMAC system consisted of a subroutine CODEVAL for the translation of an expression into machine code for evaluation. CAMAL (Barton 1971) was used in a similar way for number-crunching purpose. However, as modern compilers advance in sophistication, and as the relative cost of computing time decreases, most modern symbolic

systems resort to alternative (c) above. For example, MACSYMA, REDUCE and SCRATCHPAD allow users to readily convert expressions into FORTRAN; FORMAC was designed as a superset of PL/1, thus allowing users to program in a hybrid symbolic-numeric mode. ALTRAN and SAC, dealing with a more limited class of expressions, go even further by providing special programs to convert expressions into efficient FORTRAN code (Feldman and Ho, 1975; Collins, 1979). In the same spirit TRIGMAN includes a special processor (Shellus and Jeffreys 1975) to handle Poisson series.

A caveat emptor need be mentioned in this kind of applications. As symbolic systems provide easier access to generating expressions for numerical applications, there is an ever increasing temptation to apply this in a 'brute force' way, which often leads to disastrous results.

(2) Symbolic integration

Integrations appear in a variety of contexts in scientific computing. In the symbolic-numeric symbiosis, we may cite three broad categories. The first involves a one-shot integration, where the result is used for subsequent numerical evaluation. As an example we may cite the application by Hanson and Phillips (1978). In the investigation of numerical solution of two-dimensional integral equations, they need to perform a Laplace inversion. There MACSYMA was used to integrate a two-dimensional kernel. A second category involves multi-dimensional integrations. Here the final goal is usually numerical evaluation, but one is interested in reducing the dimensionality of integration as much as possible, because multiple quadrature is costly both in computing time and accuracy. Some of these examples are found in quantum electrodynamics (Fox and Hearn, 1974) and magneto-hydrodynamics (Kerner and Grimm, 1975). A third context concerns multi-parameter studies, where the integral (single or multiple) depends on a number of parameters, thus making numerical results difficult, if not impossible, to represent. For example, Feldman (1974) described such an application of ALTRAN to crystal physics. In order to tackle the variety of integration problems, Ng (1977) advocated a number of approximate schemes. For brevity these schemes will be referred to as approximations by basis functions, by interpolation and by reduction of transcendence. We believe that in the arena of symbolic integrations, a large number of hybrid methods deserve further exploration.

(3) Finite element calculations

Finite element methods have been used quite extensively and successfully in structural engineering and are finding new applications in other technical areas. Here one typically solves an elliptic boundary-value problem by the choice of a set of approximating basis functions together with the application of a variational technique. As examples we may cite several applications that involve both a symbolic and a numeric phase. A first example concerns a static problem

in structural engineering investigated by Andersen and Noor (1977) who used MACSYMA for differentiation, integration and the eventual production of FORTRAN expressions for the incorporation into a program to perform numerical computation. The authors had also devised a systematic scheme for the simplification of symmetric sub-expressions. The second example involves a dynamic problem in structural engineering studied by Gupta and Ng (1977) who used MACSYMA to perform similar functions. However, this problem is nonlinear in nature and is not an easy candidate for systematic simplification. A third example concerns a singular perturbation problem investigated by Miranker and Yun (1974). In this case SCRATCHPAD was used to derive the algebraic equations defining the coefficients in the finite element approximation. In all these applications the goal in the symbolic phase is to generate expressions for some stiffness and mass matrices which are used in turn for numerical computation. In the former two cases the authors went one step further towards generating actual FORTRAN expressions. It is safe to conjecture that this is one area where a numeric symbolic symbiosis is going to be quite fertile.

(4) Symbolic differentiation and Jacobian computation

The need for gradient and Jacobian calculation arises most often in the numerical solution of nonlinear equations, where some version of iteration is typically used. In the context of this application, one needs to read in the definition of the nonlinear equations, then symbolically differentiates the functions and expresses the partial derivatives in some high level language representations, such as FORTRAN. This type of nonlinear problem may come up in a variety of versions, such as those in optimization problems, in boundary-value problems, in stiff initial-value problems. The need of Jacobian computation may also come up in entirely different contexts. For example, in sensitivity analysis of numerical solution of differential equations, and in coordinate transformations applied to systems of equations. In the context of such numerical applications, one would eventually express the Jacobian matrix inside a program written in a high level language, such as FORTRAN or P1/1. (We use the term Jacobian matrix to subsume the special case of a gradient vector.) For example, in the design of a package of programs for optimization problems, one either assumes a user-provided program for the Jacobian or relies on derivative-free methods, which usually refer to some approximations of numerical differentiation (Brown, et al., 1976). Recognizing this need, at least two specialized packages have been developed for the generation of the Jacobian matrix in FORTRAN (Warner 1975; Kedem 1976).

At first glance this application can be handled readily by most present symbolic algebra systems. It seems that one need only to read in the function vector and to perform partial differentiation, and then to express the results in some high level language for numerical computation. However, there is a

difficulty of keeping track of the intermediate variables, and also preventing a detailed evaluation of these variables which serve as symbols for common sub-expressions. Moses (1977) introduced the name 'shadow variables' to describe these. Recently Ng and Char (1979) demonstrated how a straightforward application of a symbolic system can lead to inefficient and often useless expressions. They have further developed a MACSYMA program to produce efficient FORTRAN code for the Jacobian matrix.

It would enlighten the SIGSAM community to see a comparison of performance between the use of the specialized packages and that of the general systems.

(5) Generation of high-order difference formulas

Another possible fruitful area for a symbolic-numeric interface may be described as the calculus of differential or difference operators. This type of calculus is often required in the generation of high-order difference formulas for the solution of differential equations. As examples we may cite both initial-value and boundary-value problems. One initial-value problem has been proposed independently by Campbell (1973) and Jenks (1976), and it concerns the generation of high-order Runge-Kutta formulas. The problem involves two steps: (a) the symbolic generation of a set of nonlinear algebraic equations of conditions, and (b) the numerical solution of these equations or the demonstration of inconsistency among a set of solutions. Attempts have been made to mechanize various stages of this procedure. For example, Rochon and Strubbe (1975) used SCHOONSCHIP and Jenks (1979) used SCRATCHPAD to mechanize a calculus of operators, and therefore to generate high-order terms for the equations of conditions. Verner (1978) made a different attempt from an entirely numerical vantage and was quite successful in making this procedure more mechanical.

Turning to a boundary-value example, we witness a rather significant application in the symbolic-numeric symbiosis. Using MACSYMA, Keller and Pereyra (1978) were able to derive high order compact difference schemes for the numerical solution of ordinary differential equations with boundary conditions. Such schemes are ones that use the least number of grid points to obtain accurate approximations to a specified order.

Another example involving both initial and boundary values is reported in this symposium (Khalil and Ulery 1979). Here the authors have devised a semi-numeric algorithm to generate families of difference approximations to the heat equation in one and two dimensions. It is, however, not clear from the paper how a symbolic system has been instrumental in bringing forth the results.

These examples do lend credence to the above-mentioned thesis of man-computer symbiosis.

(6) Symbolic massaging for function approximation

Numerical computation of transcendental functions often requires preliminary experimentation with various functional forms. Certain forms that are mathematically elegant may be computationally ill-conditioned. Thus a number of activities involve analytic 'massaging' for best computational representations. For example, in the design of a Bessel function subroutine, Cody et al. (1977) needed a special series for the function and resorted to ALTRAN for aid. Fullerton and Rinker (1976) also used ALTRAN to develop rational approximations to a special function in a physics application. A number of MACSYMA applications also will bear fruit in the area of function approximation: the investigation by Avgoustis (1977) on Laplace transforms, by Gosper (1977) on hypergeometric sums, by Ng and Polajnar (1976) on elliptic integrals, and by Cuthill (1977) on the approximations of the exponential. Earlier Juncosa (1972) used FORMAC for a study of a multivariate probability function.

(7) Aid in error analysis

It would seem that symbolic systems are attractive tools to aid in the development of an ϵ -calculus for error analysis in numerical computations. In the early part of the decade Gentleman (1971) suggested an application in an error analysis of Goertzel's method. Loos (1972) applied REDUCE successfully in that application. Later Kahan (1974) used SCRATCHPAD for some of his work on error analyses. More recently Stoutemyer (1977) attempted a more systematic investigation of automatic error analysis with REDUCE. We believe this is one fertile territory where symbolic tools can be applied for significant results. As a challenge we cite Olver's new approach (1978) to error arithmetic as a candidate for implementation.

(8) Perturbation and asymptotic calculations

We have already seen that symbolic tools were primarily used in developing perturbation theories in celestial mechanics and quantum electro-dynamics. As perturbation continues to enjoy a central place in scientific theories, we think symbolic tools will find more applications in that area. This phenomenon may be viewed as a corollary to the Birkhoff theory of the epistemological stature of approximation. However, to date there has been a paucity of attempts towards general-purpose symbolic tools in automating perturbation procedures, such as the WKB approximation. Recent attempts by Fateman (1976) and Stoutemyer (1975) in the related area of asymptotic analyses are positive beginnings of more general purpose symbolic-numeric tools in such applications.

(9) Computational complexity

In the above mention of Traub's definition of numerical mathematics, one of the four components cited concerns foundations of the field. The study of the finite-precision number system and that of computational complexity are central to the foundation of numerical mathematics. In this decade we have witnessed a burst of activities in the subject of complexity and a number of

these activities and related ones can and will lead to much cross-fertilization between symbolic and numerical mathematics. Up to 1975 a number of these activities was summarized in the book of Borodin and Munro (1975). More recent research may be found in the work of Brent and Kung (1976), Brent and Traub (1978), Cohen and Katcoff (1977), Gustafson and Yun (1979), Gosper (1977), Ivie (1977), Lipson (1976), Moenck and Certer (1979), Stoutemyer (1979) and Yun (1976,1977). Some of the possible mathematical tasks either mentioned in or implied by these papers are:

- (a) the discovery of closed-form representations for finite sums of algebraic expressions;
 - (b) closed form solutions to recurrence relations;
 - (c) power series for differential equations;
 - (d) asymptotic approximations of complex expressions;
 - (e) fast iteration applied to symbolic computation;
 - (f) rational function integration;
 - (g) rational interpolation;
 - (h) polynomial arithmetic.
- (10) Qualitative analysis

In a typical symbolic, numerical or scientific computation, it is always useful to know some broad qualitative properties of the expressions or equations in question. Stoutemyer has developed a number of tools for such qualitative analysis. One program (Ref. 68) allows a user to investigate mathematical properties of analytic expressions, such as extrema, convexity, symmetry and periodicity. Another program (Ref. 69) allows a scientific user to perform dimensional analysis. A third program (Ref. 65) assists in the automation categorization of optimization and mathematical programming problems. Stoutemyer's contributions in this area will no doubt aid in the further promotion of the symbolic-numeric interface.

VI. The Power and Limitation of Analogues

Dissimilar to Birkhoff's usage of the word in Section II, we use analogue here to see what role analogy plays in the symbolic-numeric symbiosis. In describing scientific methodology, Hermann Weyl, in his "Philosophy of Mathematics and Natural Science" (Ref. 87), stated the principle of analogy as playing an important role in scientific progress. He cited Newton's emphasis in this principle and then mentioned Maxwell's elaboration on it: "By a physical analogy, I mean that partial similarity between the laws of one science and those of another which makes each of them illustrate the other." (P. 163)

As we examine the cross traffic between symbolic and numerical computation, it is natural to wonder if the principle of analogy has played a significant role in the advancement of the two fields. Since numerical

computation has a giant head-start, it is unreasonable to expect that it would adopt any methodology from the other field. Therefore we turn to the other side of the coin. Here we do see a few instances of successful use of analogy. We cite some particularly illustrative examples. A most obvious analogy is between integer and polynomial arithmetic, as expounded in Aho, Hopcroft and Ullman (1974). The Fast Fourier Transform is in turn closely related to polynomial multiplication and has been useful for the construction of hybrid algorithms for fast multiplication (Moenck 1976). The recent advance in FFT (Winograd 1979) will no doubt stimulate further progress in this part of the symbolic-numeric interface. Another area of active research revolves around polynomial zero-finding. Here the symbolic mathematician is at once reminded of the heritage he shares with the numerical mathematician. The most recent survey of this problem is given by Collins (1977) from the vantage of a symbolic mathematician and by Jenkins and Traub (1975) from that of a numerical mathematician. Turning to another active area of research, linear algebra, we witness a very high level of sophistication in numerical computation (cf. Stewart 1973 and LINPACK 1979). By contrast, however, the symbolic counterpart is considerably more primitive, though the work by Cabay (1977) and McClellan (1977) on numeric data, and by Griss (1976) and Wang (1977) on symbolic data constitute at least a promising start towards a sophisticated 'symbolic linear algebra'. Switching our attention from linear to nonlinear equations, we are at once reminded of the pervasive influence of Newton's and related methods of iterations in numerical mathematics. Many a symbolic mathematician has recently found this class of primarily numerical technique to be quite useful in the manipulation of power series, rational and algebraic functions (Kung and Traub 1978; Lipson 1976; Moenck 1979; Yun 1975; Yun 1976). Though the above list is far from comprehensive, it does indicate the flavor of how symbolic mathematicians have been taking advantage of numerical techniques and apply the variance of these techniques to symbolic computation. In much of the work mentioned in this section, we again see the pivotal role of approximation, whether it be direct or iterative. We believe the rich heritage of numerical mathematics presents an important cornucopia of concepts for the symbolic mathematician to develop new tools.

VII. Epilogue

Since this paper is presented to a SIGSAM audience, it is written from the vantage of symbolic mathematics. It is interesting to note that 30 years ago Herman Weyl (Ref. 81) used this last phrase in a different context, to distinguish from intuitive mathematics. As we survey the two-way traffic between numerical and symbolic computation, we may offer the following conclusions. (a) Most scientific computation consists of an analytic and a numerical phase. (b) In the past decade

there has been a modest beginning of symbiotic approach. (c) All successful symbiotic (numeric-symbolic) applications depend on another symbiotic (man-machine) cooperation. (d) To date there has been some applications of numerical mathematics research to symbolic mathematics research and less in the other direction. Much more potential can be exploited. (e) There are a number of areas quite ready for more research efforts. In other words, the symbolic-numeric interface offers an interesting horizon for research.

VIII. A Log of Acknowledgements

When I first conceived of this survey, I was not aware of the paper by Brown and Hearn. After I wrote to a number of colleagues, Brown and Hearn sent me their preprints. Thus the present survey is written in a format complementary and supplementary to their work. I am further indebted to Caviness, Jenks, Stoutemyer, Traub, van Hulzen, and Yun who sent me useful information and preprints, and specifically to Drs. van Hulzen and Yun who pointed out to me some omissions in the earlier manuscript. This work is supported in part by the National Science Foundation and in part by the National Aeronautics and Space Administration.

References

1. A. V. Aho, J. E. Hopcroft, and J. D. Ullman, The Design and Analysis of Computer Algorithms, chap. 8, Addison-Wesley: Menlo Park, California (1974).
2. C. H. Andersen and A. Noor, "Symbolic Manipulation Techniques for Vibration Analysis of Laminated Elliptic Plates," Proc. of 1977 MACSYMA User's Conference, NASA Publication #CP-2012, NASA, Washington, D. C. (1977).
3. Y. Avgoutis, "Symbolic Laplace Transforms of Special Functions," Proc. of 1977 MACSYMA Users' Conference, NASA Publication #CP-2012, NASA, Washington, D. C. (1977).
4. D. Barton, I. M. Willers, R. Zahar, "Taylor Series Methods for ODE," in Mathematical Software, ed. J. Rice, Academic Press, N. Y. (1971).
5. G. Birkhoff, "Mathematics and Psychology," SIAM Rev. Vol. 11, pp. 429-469 (1969).
6. A. Borodin and I. Munro, The Computational Complexity of Algebraic and Numeric Problems, American Elsevier, N. Y. (1975).
7. R. P. Brent and H. T. Kung, "Fast Algorithms for Manipulating Formal Power Series," Dept. of Comp. Sci. Rept., Carnegie-Mellon Univ. (January 1976).
8. R. P. Brent and J. F. Traub, "On the Complexity of Composition and Generalized Composition of Power Series," Dept. of Comp. Sci. Rept., Carnegie-Mellon Univ. (May 1978).
9. W. S. Brown and A. C. Hearn, "Applications of Symbolic Algebraic Computation," Comp. Phys. Comm. to appear.
10. S. Cabay and T. P. L. Lam, "Congruence Techniques for the Solution of Integer Systems," ACM Trans. Mathematical Software, Vol. 3, pp. 386-397 (1977).
11. J. A. Cambell, ACM/SIGSAM Bulletin #25, pp. 13-14 (1973).
12. W. J. Cody, Rose M. Motley, and L. Wayne Fullerton, "The Computation of Real Fractional Order Bessel Functions of the Second Kind," ACM Transactions on Mathematical Software 3, pp. 232-239 (September 1977).
13. J. Cohen and J. Katcoff, "Symbolic Solution of Finite-Difference Equations," ACM Trans. Mathematical Software, Vol. 3, pp. 261-271 (1977).
14. G. E. Collins, "Infallible Calculation of Polynomial Zeros to Specified Precision," Mathematical Software III, ed. J. R. Rice, Academic Press, N. Y. (1977).
15. G. E. Collins and S. C. Schaller, SAC1 User's Guide and SAC1 Implementation Guide, Comp. Sci. Tech. Repts. #268 and #269, Univ. of Wisconsin (1976). (A SAC2 manual is forthcoming in the same series of reports.)
16. F. D. Cray, "The Augment Precompiler: I. User Information," U. of Wis., Math. Res. Ctr. Rept. #1469 (December 1974).
17. E. Cuthill, "Rational Approximations to e^{-x} with Negative Real Poles," Proc. of 1977 MACSYMA User's Conference, NASA Publication #CP-2012, NASA, Washington, D. C. (1977).
18. J. J. Dongarra, J. R. Bunch, C. B. Moler, and G. W. Stewart, LINPACK Users' Guide, Soc. Indus. Appl. Math. Publications, Philadelphia (1979).
19. EUROSAM '74, ACM/SIGSAM Bulletin, Vol. 8 #3 (1974).
20. R. J. Fateman, "An Approach to Automatic Asymptotic Expansions," Proc. of 1976 ACM Symp. on Sym. & Alg. Comp., pp. 365-371, ACM:NY (1976).
21. R. J. Fateman, "An Improved Algorithm for the Isolation of Polynomial Real Zeros," Proc. of 1977 MACSYMA Users' Conference, NASA Publication #CP-2012, NASA, Washington, D. C. (1977).
22. S. I. Feldman, "An Application of Symbolic Computation to Crystal Physics," Computing Science Technical Report No. 15, Bell Laboratories, Murray Hill, N. J. (May 1974).
23. Stuart I. Feldman and Julia Ho, "A rational Expression Evaluation Package," Computing Science Technical Report No. 34, Bell Laboratories, Murray Hill, N. J. (September 1975).
24. J. A. Fox and A. C. Hearn, "Analytic Computation of Some Integrals in Fourth Order Quantum Electrodynamics," J. Comp. Physics 14, pp. 301-317 (March 1974).
25. L. Wayne Fullerton and G. A. Rinker, Jr., "Accurate and Efficient Methods for the Evaluation of Vacuum-Polarization Potentials of Order $Z\alpha$ and $Z\alpha^2$," Physical Review A 13, pp. 1283-1287 (March 1976).

26. W. M. Gentleman, "A New Problem Section for the Bulletin," ACM/SIGSAM Bulletin, pp. 4-6 (April 1971).
27. R. W. Gosper, Jr., "Indefinite Hypergeometric Sums in MACSYMA," Proc. of 1977 MACSYMA Users' Conference, NASA Publication #CP-2012, NASA, Washington, D. C. (1977).
28. M. Griss, "The Algebraic Solution of Large Sparse Systems of Linear Equations Using REDUCE2," Proc. of ACM National Conference, Vol. 1, pp. 105-111 (1974).
29. M. L. Griss, "The Algebraic Solution of Sparse Linear Systems via Minor Expansion," ACM Trans. Mathematical Software, Vol. 2, pp. 31-49 (1976).
30. K. Gupta and E. W. Ng, "Symbolic Calculations in a Finite Dynamic Element Analysis," Proc. of 1977 MACSYMA Users' Conference, NASA Publication #CP-2012, NASA, Washington, D. C. (1977).
31. F. G. Gustavson and D. Y. Y. Yun, "Fast Computation of the Rational Interpolation Table and Toeplitz Systems of Equations via the Fast Extended Euclidean Algorithm," these Proceedings (1979).
32. A. C. Hearn, ed., REDUCE Newsletter, Univ. of Utah, Symbolic Comp. Grp., Dept. of Comp. Sci. (1978).
33. J. Ivie, "Some MACSYMA Programs for Solving Difference Equations," MACSYMA Users' Conf. Proc., pp. 317-326, NASA Publication #CP-2012, NASA, Washington, D. C. (1977).
34. M. A. Jenkins and J. F. Traub, "Principles for Testing Polynomial Zerofinding Programs," ACM Trans. on Mathematical Software, Vol. 1, pp. 26-34 (1975).
35. R. D. Jenks, ACM SIGSAM Bulletin #37, pp. 6-7 (1976).
36. R. D. Jenks, "SCRATCHPAD/360: Reflections on a Language design," ACM/SIGSAM Bulletin, Vol. 13, #1 (February 1979).
37. M. Juncosa, "A General Analytic Approach ...," Information Processing 71, pp. 1356-1362, Proc. of IFIP Congress 1971, North Holland Publishing Co. (1972).
38. W. H. Kahan, "One Numerical Analyst's Experience with One Symbol Manipulator," SIAM Rev. Vol. 16, #1, p. 129 (1974).
39. G. Kedem, "Auto. Diff. of Comp. Programs," Univ. of Wis. Matr. Res. Ctr. Report #1697 (November 1976).
40. H. Keller and V. Pereyra, "Symbolic Generation of Finite Difference Formulae," Math. Comp. Vol. 32, pp. 955-971 (1978).
41. W. Kerner and R. C. Grimm, "Magneto-hydrodynamic Spectra for Tokamaks with Non-circular cross Sections," Proceedings of the Seventh Conference on Numerical Simulation of Plasmas, Courant Institute, N. Y. (June 1975).
42. D. Knuth, "Seminumerical Algorithms," The Art of Computer Programming, Vol. II, pp. 360-443, Addison-Wesley: Reading, Mass. (1969).
43. H. T. Kung and J. F. Traub, "All Algebraic Functions Can Be Computed Fast," J. of ACM, Vol. 25, pp. 245-260 (1978).
44. J. C. R. Licklider, "Man-Computer Symbiosis," IRE TRANSACTIONS on Human Factors in Electronics (March 1960).
45. J. D. Lipson, "Newton's Method: A Great Algebraic Algorithm," SYMSAC 76, pp. 260-270 (1976).
46. Rudiger Loos, "Solution of Problem Number 1 by REDUCE2," ACM SIGSAM Bulletin, pp. 32-40 (January 1972).
47. MACSYMA Users' Conf. Proc., NASA Publication #CP-2012, NASA, Washington, D. C. (1977).
48. M. T. McClellan, "A Comparison of Algorithms for the Exact Solution of Linear Equations," ACM Trans. Mathematical Software, Vol. 3, pp. 147-158 (1977).
49. W. L. Miranker and D. Y. Y. Yun, "A Note on Boundary Layer Elements," IBM Research Note #RC 4990, T. J. Watson Res. Ctr., Yorktown Heights, N.Y. (1974).
50. R. Moenck, "Practical Fast Polynomial Multiplication," SYMSAC 76, pp. 136-145 (1976).
51. R. Moenck and J. Carter, "Approximate Algorithms to Derive Exact Solutions to Systems of Linear Equations," these Proceedings (1979).
52. J. Moses, "The Variety of Variables in Mathematical Expressions," Proc. of 1977 MACSYMA Users' Conf., NASA Publication #CP-2012, NASA, Washington, D. C. (1977).
53. E. W. Ng and B. Char, "Gradient and Jacobian Computation for Numerical Applications," Proc. of Second MACSYMA Users' Conference, Washington, D. C. (1979).

54. E. W. Ng and D. Polajnar, "A Study of Alternative Methods for the Symbolic Calculation of Elliptic Integrals," SYMSAC 76, pp. 372-376 (1976).
55. E. W. Ng, "Observations on Approximate Integrations," Proc. of 1977 MACSYMA User's Conf., NASA Publication #CP-2012, NASA, Washington, D. C. (1977).
56. F. W. J. Olver, "Asymptotics and Special Functions," Academic Press: N. Y. (1974).
57. F. W. J. Olver, "A New Approach to Error Arithmetic," SIAM J. of Num. Anal. Vol. 15, pp. 368-393 (1978).
58. J. R. Rice, "Prologue," in Mathematical Software, ed. J. Rice, Academic Press, N. Y. (1971).
59. J. R. Rice, "On the Present and Future of Scientific Computation," Comm. ACM, Vol. 15, pp. 637-639 (1972).
60. M. Rochan and H. Strubbe, ACM/SIGSAM Bulletin #36, pp. 30-38 (1975).
61. J. E. Sammet, "Software for Nonnumerical Mathematics," in Mathematical Software, ed. J. R. Rice, Academic Press, N. Y. (1971).
62. P. J. Shelus and W. H. Jeffreys, III, "A Note on an Attempt at More Efficient Poisson Series Evaluation," Celestial Mechanics 11, pp. 75-78 (1975).
63. SIGSAM Bulletin, Vol. 12, #2, Assoc. for Comp. Mach. (1978).
64. G. W. Stewart, Introduction to Matrix Computations, Academic Press, N. Y. (1973).
65. D. R. Stoutemyer, "Analytical Optimization Using Computer Algebraic Manipulation," ACM Transactions on Mathematical Software 1, pp. 147-164 (June 1975).
66. D. R. Stoutemyer, "Automatic Error Analysis Using Computer Algebraic Manipulation," ACM Transactions on Mathematical Software 3, pp. 26-43, (March 1977).
67. D. R. Stoutemyer, "Automatic Categorization of Optimization Problems," Operations Research, to appear.
68. D. R. Stoutemyer, "Qualitative Analysis of Mathematical Expressions ...," Proc. of ACM Symp. on Sym. & Comp., pp. 105-113, ACM, N. Y. (1976).
69. D. R. Stoutemyer, "Dimensional Analysis, Using Computer Symbolic Mathematics," J. of Comp. Phys. Vol. 24, pp. 141-149, (1977).
70. SYMSAM II, Proc. of 2nd Symp. on Sym. and Alg. Manipulation, ed. S. Petrick, ACM, N. Y. (1971).
71. SYMSAC 76, Proc. of A Symp. on Sym. and Alg. Comp., ACM, N. Y. (1976).
72. R. G. Tobey, "Symbolic Mathematical Computation - Introduction and Overview," SYMSAM II, p. 10 (1971)
73. J. R. Traub, "SIGNUM Panel on the Future of Numerical Mathematics," ACM/SIGNUM Newsletter, Vol. 7, pp. 40-45 (1972).
74. J. F. Traub, "Numerical Mathematics and Computer Science," Comm. ACM, Vol. 15, pp. 537-541 (1972).
75. J. F. Traub, ed., Analytic Computational Complexity, Academic Press: N. Y. (1975).
76. J. F. Traub, ed., Algorithms and Complexity, Academic Press: N. Y. (1976).
77. D. L. Ulery and H. M. Khalil, "A Semi-Numeric-Difference Algorithm," these Proceedings (1979).
78. J. Verner, "Explicit Runge-Kutta Methods with Estimates of the Local Truncation Error," SIAM J. of Numerical Analysis Vol. 15, pp. 772-790 (1978).
79. P. Wang, "Matrix Computations in MACSYMA," MACSYMA Users' Conf. Proc., pp. 435-446, NASA Publication #CP-2012, Washington, D. C. (1977).
80. D. D. Warner, "A Partial Derivative Generator," Computing Science Technical Report No. 28, Bell Laboratories, Murray Hill, N. J. (April 1975).
81. H. Weyl, "Philosophy of Mathematics and Natural Science," pp. 162-163, Princeton Univ. Press, Princeton (1949).
82. S. Winograd, "On Computing Discrete Fourier Transform," these Proceedings (1979).
83. D. Y. Y. Yun, "Algebraic Algorithms Using P-adic Constructions," SYMSAC 76, pp. 248-259 (1976).
84. D. Y. Y. Yun, "Fast Algorithm for Rational Function Integration," Information Processing 77, North-Holland, Amsterdam (1977).
85. D. Y. Y. Yun, "Hansel Meets Newton - Algebraic Construction in an Analytic Setting," Analytic Computational Complexity, ed J. Traub, Academic Press, N. Y. (1975).