My View of the Future of Symbolic and Algebraic Computation

Richard J. Fateman
University of California, Berkeley

1. Introduction

This note has been composed in response to Bruno Buchberger's invitation to the program committee of Eurocal '85 (and others), to provide a short paper on this topic.

2. Workstations

I expect that in the near term, the most apparent change in the the nature of symbolic computation is the micro-computer and personal computer impact: the immediacy of graphics and bit-map displays, coupled with the power of large-address space chips, will make it possible to provide the most powerful of the existing algebraic manipulation systems on inexpensive machines (under $5,000) very shortly. At UC- Berkeley, we now have a complete version of MACSYMA running on several different Motorola 68000-based graphics workstations. The real-time performance level is better than any we have experienced on time-sharing systems, and the virtual memory (16 megabytes) of these systems provides ample rooom for computation.

Of course, continued experimentation on new systems, such as Maple at the University of Waterloo, can capitalize on this trend without necessarily adopting the baggage of older systems.

Independent algebraic computations of the sort that fester for long periods of time, and are consequently unwelcome inhabitants of time-shared systems, also present opportunities for these small machines which typically have reasonably fast integer and memory operations.

3. Graphics

The advent of low-cost bit-map graphics, even to the level of the least expensive personal computers, will have an enormous impact on the availability of higher-resolution graphics, on the slightly more expensive computers. Thus, 800 by 1000 monochrome resolution is likely to become fairly commonplace. This provides an opportunity for typeset-quality display of conventional equations, the introduction of special symbolic notations, and, with suitable pointing devices, the use of selection from expressions and menus. This is the situation at UC- Berkeley, and we expect this to propagate to the commercial world in the near future.

4. Data Abstraction and Mathematics

Efforts at several research sites have been leading to the cleaning up of systems for algebraic abstraction and practical computation. At UC- Berkeley, a group of researchers are working on the Berkeley Algebra and Analysis Development: a hierarchical collection of programs structured along the lines of modern algebra and analysis, all written in a language (built, for the moment,, on Lisp), called Newspeak. Newspeak [ref: J. Foderaro, PhD dissertation, UC Berkeley] provides a type-checking framework, and an inheritance structure which is designed specifically for algebraic manipulation system building. An effort of similar intent has been the Newspad project at IBM.

5. Expertise

Prolog advocates have entered the field of algebraic manipulation. I do not believe that what this field has been missing is unification as a control structure. Nevertheless, the idea of using data-base ideas, a separate and documented data base with access mechanisms, is clearly of some merit. SMP's library is such a data base. By contrast, most of the information in MACSYMA is bound up in hand-coded programs (e.g. for simplication of trigonometric functions). Much of this information, in contrast to the algebraic ideas, is not as easily developed as a hierarchy: there are just not enough universal structures for analysis and applied mathematics to organize the material we have already seen in programs.

I expect rather impressive systems to be made available by dint of hard work by applications programmers. The programs will ordinarily combine some aspects of algebraic manipulation with numerical calculation, plotting, or other external indications of the answers. Applications may frequently require the communication of results between computers (e.g. a symbolic processor and a highly parallel numerical processor.).

6. Education

Symbolic computation systems may have an impact on education, eventually. At least in this country, I do not see that happening for some years, even on the college level. Almost without exception, the spread of computers to students has been based on cost, restricting the machinery to systems which can support word-processing and numerical computation, but only very limited symbolic computation. When machines with 2 to 4 megabytes of main memory become commonly available, there will be more impact. I have written off most of my academic colleagues in this: educators do not see any need for teaching or even mentioning the Risch algorithm in Freshman calculus; and mature researchers, for the most part, have attained their distinction without the use of symbolic computation, and resist computers. We must wait yet another academic generation. My students who have used Macsyma in doing their physics and statistics homework recognize the usefulness. When the cost is low, they will take advantage of the opportunity.

7. Algorithms and Parallelism

I note that some of my colleagues have broadened the scope of symbolic and algebraic computation to include most of computer science, and some areas on the mathematical fringes. I am not convinced that there are any good new ideas for parallel symbolic algorithms. There are a few old ones based on parallel concepts which have occurred in other contexts (e.g. FFT). Speedups based on multiple parallel modular computations must rely on large numbers of moduluar images being needed, but there are few such algorithms of practical interest. Most need one, two, or perhaps as many as six, to take care of all but astronomically hard and thus uninteresting cases. For parallelism to be successfully applied, the small, common cases must be solved fast.

8. Artificial Intelligence

I have already touched on this in the section on expertise. There is a hope that algebraic computation represents an area of intelligent behavior which could serve as a model for the rigorous construction of more powerful intelligent behavior on a larger scale. I see only a few glimmers that there is this type of link-up between AI and mathematics, and the glimmers are not explicitly recognized in symbolic computation.

Heuristics and pattern-matching have been used only in very limited contexts. Planning and goal-driven automatic transformations have not had any impact on symbolic computation. There have been only a few, rather weak, special demonstrations of planning. It is clear that for some program to simulate the general mathematical problem-solving ability of a good applied mathematician, programs would have to be orders of magnitude more capable than they are now. It is my belief that progress can be made by symbolic computation specialists providing suitable languages and interfaces to the application-development specialists. In exchange, such applications can be used by AI advocates to demonstrate truly impressive behavior reflected a model of mathematics and a physical domain, rather than a mere collection of ad-hoc statement of preferred responses to particular anticipated situations.

9. Acknowledgments

This work was supported in part by DOE Contract DE-AM03-76SF00034, P. A. DE-AT03-76E10358, and in part by the System Development Foundation.