"Intelligent Techniques in the Solution of Nonlinear Differential Equations"

Richard L. Peskin
Rutgers University
CAIP Center Box 1390 CoRE Bldg.
Piscataway, NJ 08855-1390
<peskin@caip.rutgers.edu>

Abstract

The integration of symbolic computation with numeric computation is a software direction that offers considerable potential, not only in terms of improved performance, but also for effecting solution verification and completeness. This note discusses issues associated with the incorporation of mathematical knowledge into the symbolic-numeric computation process. The problem considered is the stiff nonlinear boundary value differential equation problem. Methods to employ symbolic computing, together with use of a specialized mathematical knowledge base allow this type of problem to be solved with much greater efficiency than with typical "brute force" numerical methods. In addition, the methodology assures identification of all stable solutions, and problem based parallelism is identified. Challenges associated with extending these concepts to other classes of problems are discussed.

Introduction

Although recent thrusts in scientific computing have emphasized hardware speed, memory size, etc., the scientific computation community is beginning to recognize that there are major gains in "computing smarter", not just bigger and faster. Thus there is more attention being paid to the role of expert systems and symbolic mathematics as a tool in numerical computation. Symbolic mathematics and expert systems have been used in automatic program generation for partial differential equations [1], and symbolic processing together with a symbolic algebra package have been used in automatic solver systems for differential equations [2]. Some other applications where symbolic - numeric methods are of value were discussed by Peskin [3]. Equation classification and algorithm generation are just two of the areas where expert systems, primarily of the deep knowledge type, have been employed. In this paper, we will be concerned with a particular application of "intelligent techniques", namely, preprocessing prior to numerical solution. The basic idea is simple; if one were able to determine a priori the approximate nature of a solution, then a great deal of numerical work associated with the solution development starting from arbitrary initial guesses could be eliminated. The implementation of this concept is not simple, and one has to be prepared to call on techniques from mathematics, computer algebra, optimization theory, and other fields. However, the benefits are extensive. They include major gains in numerical computation speedup, better identification of meshing requirements, and, perhaps most important, identification of all of the stable solutions, not just a particular one which results from some arbitrary initial condition choice.

An Example Problem

Techniques for numerical solution of nonlinear differential equations have been well studied, and there are many established solver implementations. This might lead one to believe that there is no need for AI or other intelligent methodologies. However, one need only examine the following class of problems (called singular perturbation problems) to see that differential equation solution is a non-trivial task:

$$\varepsilon\frac{d^2y}{dx^2}=F(y-a,x)G(y-b,x)H(y-c,x)$$
$$\frac{dy}{dx}(0)=p,\ \frac{dy}{dx}(1)=q$$

In the above equation, ε is a small parameter; F, G, H are functions and p, q fixed (boundary) values. This type of problem occurs in any number of complex scientific applications. A typical one is fluid dynamic flow with boundary layers and shock waves, another involves chemical reactions. In the limit ε -> 0, one has a, so called, singular perturbation. In that limit, the vanishing of the highest order derivative prevents the outer or fixed point solutions:

$$0=F(y-a,x)G(y-b,x)H(y-c,x)$$

from satisfying both boundary conditions. For ordinary differential equations such as listed above, there problem has received considerable attention by mathematicians. It is known that any solution that is stable must be composed of the stable sections of the fixed point solutions; $F = 0$, $G = 0$, $H = 0$, joined together by "connecting" functions that attach the fixed point solutions to the boundary values and to each other. (In the former case the connection is called a boundary layer; in the latter it is a shock. In certain cases the connection joins derivatives of the fixed point solutions and this is called a corner layer. Each of these layers has physical interpretation.)

One interesting feature of this problem is that there is not necessarily a unique solution; any solution joining the stable regions to each other and the boundary values is admissible. In the above example, three solutions are possible. A "brute force" numerical approach would, most likely, find only one solution. Another numerical problem occurs with the need to identify locations of the "singular" layers, for these are the regions where most rapid changes occur and, thus, require the greatest numerical resolution. While the location of the boundary layers may be obvious, locations of shock or corner layers is not. It is inefficient to impose fine resolution over the whole domain when such resolution is required only over a small portion of the domain. Additionally, numerical techniques that are based on initial guesses can be very consumptive of computing resources just to relax the initial guess to something approaching the actual solution.

Current Research Directions

For the past three years we have been developing intelligent (e.g. AI) computation techniques to address these problems for both ordinary and partial nonlinear differential equations. [3,4,5] Our approach has been to incorporate expert system and path searching technology into numerical solvers with the objective of: (a) identifying the stable regions of the fixed point solutions to be connected, (b) approximating the functional form of the "connectors", (c) identifying all possible stable paths joining connectors and fixed point solutions, (d) using a,b,c to locate the singular layers in advance as well as specifying an appropriate initial approximation for each possible solution. An additional benefit from the use of AI techniques is the identification of levels of parallelism inherent in the solution process.

The intelligent techniques that we use are based on the computational instantiation of mathematical results from the study of the theory of differential equations. The first challenge is to develop equivalent localized versions of the relevant mathematical theorems; the global forms developed by mathematicians are not well suited for computer implementation. Once this was done, the computational tasks to establish a,b,c,d, above were implemented. (a) requires that each fixed point solution be satisfy a set of partial differential inequalities of the form,

$$\frac{\partial^{2q+1} R(u(x),x)}{\partial u^{2q+1}} \geq m > 0$$

where R represents the right hand side of the original equation ($F*G*H$), and u is a fixed point solution candidate. These inequality relations (which form part of our expert system) determine the range of stable solutions for several classes of stability. (See the paper by Russo and Peskin [4] for details.) We have employed a computer algebra system (Maple) to do this task. As part of this task the algebra system also solves for the fixed point solutions. However, this sort of task is greatly stressing the capabilities of commercially available computer algebra systems, and other techniques to deal with these partial inequalities are under investigation.

Once the stable regions are found, sufficient parameters are available to do (b). The mathematical theorems provide bounding functional relations that can be used to determine "connectors"; these relations are part of the expert system. Again the computer algebra system can be used for simplification and evaluation. Given all the stable regions and connector definitions, the determination of paths (c) can be effected by "usual" AI path finding techniques. Fortunately, most practical problems do not admit so many possible solutions that this is an onerous computational task for ordinary differential equations, however, it may be non-trivial for partial differential equations. Finally (d) is accomplished by collecting each complete path that joins the boundary values and stable regions of the fixed point solutions. Assuming the computer algebra system is able to provide a symbolic solution for the fixed points, then each admissible path is symbolically determined. (Otherwise we may have a mixed symbolic numeric representation.) Each path forms a complete initial guess for the succeeding numerical processing. In addition, the location of singular layers is known, so that localized grid refinement can be used.

This procedure (AI based mixed symbolic numeric solution) provides several benefits. The initial solutions identified (symbolically) are very close to "final" solutions. This implies very large overall speedup for the numerical portion of the process. The locating of singular regions also results in more efficient numerical computation. Of importance is the ability of this methodology to identify ALL stable solutions, not just one which is a result of arbitrary initial condition choice. Furthermore, the symbolic initial solution can be used to gauge the viability of numeric results.

The methodology results in identification of the following large grain parallelization:

Numerical solution for each admissible solution can proceed in parallel, (2) Numerical solution for any given path can be done in parallel because the solutions in the vicinity of the singular regions are essentially independent of each other (since they are connected through the know fixed point solutions). (3) Symbolic processing to determine stable regions for each fixed point solution can be done in parallel.

Challenges

There are a number of challenging problems to be studied as we advance this mixed symbolic numeric approach. The heavy reliance on computer algebra systems has begun to tax their current capabilities. These systems (Mathematica, Maple, etc.) were designed to service more general user needs. While traditionally emphasizing algebraic operations such as GCD determination, later versions now provide many numerical services, such as differential equation solution. But solution of stiff boundary value problems is difficult, if not impossible using most of the commercial computer algebra packages. We have found limitations like excessive expansion of terms to be more severe as we deal with more complex problems such as nonlinear partial differential equations. Of particular concern has been the ability of these systems to deal with the simultaneous partial differential inequalities required for determination of stability. We are currently looking at alternatives to commercial algebra packages. One possible approach is the use of optimization algorithms [6]. While symbolic solution is most desirable, some of the newer parallel optimization packages may be able to provide accurate solutions rapidly. Assuming we continued to use pure symbolic methods, clearly there is a need for parallel symbolic computation methods. While we can identify concurrent symbolic tasks, at present we can only effect these as distributed. That is, although our methodology identifies parallel symbolic tasks, there is a need for parallel symbolic packages that can take advantage of such identification. There is a need to "standardize" the mathematical expert components so that the methodology can be made more portable. This particularly true as we expand the class of equations handled. One of the most important challenges comes in expanding this overall approach to other classes of equations, including partial differential equations. The methodology clearly rests on a strong foundation of fundamental mathematical knowledge. The success of applying our intelligent techniques depends on the existence of the necessary mathematical knowledge base. This knowledge base becomes considerable more sparse as equation complexity increases. We are currently experimenting with the application symbolic - numeric methods to classes of differential equations which exhibit bifurcation behavior. But we are no where near being able to apply these methods to complex systems such as the Navier-Stokes equations. Given the difficulty of establishing firm mathematical results for nonlinear systems, such that those results could be incorporated into knowledge bases, one is tempted to look at other approaches to access the needed mathematical information. We are currently researching the use of graphical based qualitative knowledge as an alternative to traditional quantitative theoretical methods.

Summary

In summary, use of AI techniques and expert system methodology as a preprocessor in differential equation solvers offers much promise. Results to date have yielded significant numerical speedup, ability to identify all possible solutions, and ability to identify parallelization for the solution process, and ability to provide a viability check for numerical results. Numerous challenges are presented, such as the need to provide computer based symbolic operations aimed primarily for numeric - symbolic use, as opposed to the more general uses which are the subject of most available symbolic packages. For immediate needs, better methods of handling differential inequalities are essential. Future enhancement of this approach may depend on advances is symbolic computational mathematics implemented themselves with expert system and AI techniques. We need to provide sophistication in symbolic computation that will be on a par with that of numeric computation.

Acknowledgments

Research sponsored by NSF Grant ECS-9110424, and by the Rutgers University CAIP Center.

References

[1] Kowalski, A.D., Russo, M.F., Peskin, R.L., "Anatomy of Agnes: An Automatic Generator of Numerical Equation Solutions", Mathematics and Computers in Simulation, 31 (1989), pp. 343-352

[2] Boubez, T.I., Froncioni, A.M. and Peskin, R.L. "A Prototyping Environment for Differential Equations", Proc. of IMACS 2nd International Conf. on Expert Systems in Numerical Computing, Intelligent Scientific Software Systems, North Holland, 1990.

[3] Peskin, R.L. "Symbolic Computation in Engineering User Interface Systems", Proc. of the Symposium on Symbolic Computations and their Impact on Mechanics, Noor, A.L., Elishakoff, I. and Hulbert, G. ed., ASME, pp. 97-111, Nov. 1990.

[4] Russo, M. F. and Peskin, R. L. "Automatically Identifying the Asymptotic Behavior of Nonlinear Singularly Perturbed Boundary Value Problems" J. of Automated Reasoning, V. 8 N. 3 , June 1992

[5] Russo, M. F. and Peskin, R. L. "Automatic Domain Partitioning for Parallel Program Generation Using Nonlinear Singular Perturbation Theory" in "Parallel Processing for Scientific Computing" SIAM, Phila., 1989 pp. 180

[6] Russo, M. F. private communication, July 1992