

FINAL  
IN-61-CR  
17 cit.  
5043  
P-1

Final & Interim Reports  
for NASA Grant NAG2-513  
The Analysis of Control Trajectories  
Using Symbolic and Database Computing

Robert Grossman  
University of Illinois at Chicago

June, 1995

(NASA-CR-199484) THE ANALYSIS OF  
CONTROL TRAJECTORIES USING SYMBOLIC  
AND DATABASE COMPUTING Final  
Report, 1 Jan. 1992 - 31 Dec. 1994  
(Illinois Univ.) 7 p

N96-10876

Unclas

G3/61 0068099

1.

SEP 05 1995  
TO CASI

## 1 Introduction

This is a final report for NASA grant NAG2-513, "The analysis of control trajectories using symbolic and database computing." This report covers the period January 1, 1992 through December 31, 1994. *This report also comprises the formal semi-annual status reports for this grant for the periods June 30–December 31, 1993, January 1–June 30, 1994, and June 1–December 31, 1994.*

The research supported by this grant is broadly concerned with the symbolic computation, mixed numeric-symbolic computation, and database computation of trajectories of dynamical systems, especially control systems. The NASA Technical Officer for this grant is Dr. George Meyer, NASA Ames Research Center, Mail Stop 210-3, Moffet Field, California, 94035. In Section 2, we provide an overview of the progress made during the report period. Section 3 contains bibliographic references for the articles related to this grant that were completed during this period.

## 2 Review of Work During Report Period

*Trajectories and Approximating Series.* Nonlinear dynamical systems, control systems, and hybrid systems all have trajectories which describe the behavior of the system as time evolves. The analysis of these trajectories and the derivations of specialized numerical algorithms which approximate them is greatly aided by the exact computation of symbolic series which approximate the trajectories. These are series involving differential operators—developing the algorithmic foundations of the symbolic analysis of such series has been an important focus for our work. Taylor series, Lie series, and Fliess-Chen series are all examples of such series.

Technically, it has turned out to be useful to start with a ring of functions  $R$  (such as polynomials or formal power series, which can be managed easily symbolically) and to view differential operators as higher order derivations of  $R$ . A derivation is simply a linear map of  $R$  to itself satisfying Leibnitz' rule, while a higher order derivation can be viewed for the purposes here as a formal symbolic expression in derivations of  $R$ .

*The Cayley algebra of trees.* Symbolic computations involving differential operators typically involve cancellations. It turns out that developing efficient algorithms for computing symbolically with differential operators depends upon exploiting these cancellations [Crouch92]. In prior work, we

developed a data structure involving trees labeled with derivations to keep track of certain of these cancellations. By doing this, one can develop more efficient algorithms for the symbolic computation of differential operators. The analysis of these cancellations is simplified by imposing an addition and multiplication on trees so that the action of the differential operators factors through an action of the algebra of trees. We call the resulting algebra the *Cayley algebra* of trees, which we denote by  $\mathcal{T}$ . It turns out that it is a Hopf algebra. These and related ideas are described in [Grossman92b], where a number of algorithms are presented for the symbolic manipulation of expressions involving derivations.

*Actions of differential operators.* An application in which the symbolic computation of derivations is important is the derivation of numerical algorithms, especially application-specific numerical algorithms. One means of deriving numerical algorithms in an invariant or basis-free fashion is to compute the action of a flow (which can be considered as a differential operator) on a test or observation function. The symbolic computation of this action requires extending the ideas above involving trees from the context of algebras to the context of modules. More precisely, we showed how to view the ring of observation functions  $R$  as a  $\mathcal{T}$ -module. These ideas are described in [Crouch91] and [Crouch93]. In [Crouch92], we specialize this calculus to systems with polynomial coefficients and show how it can be used to derive first integrals for such systems. In [Doffou95], we specialize this calculus to derive differential invariants for vector field systems with polynomial coefficients. This provides an explicit symbolic algorithm for a problem which has been outstanding for many years. In [Grossman94b] we applied closely related ideas to study normal forms of control systems.

*Geometrically stable integration algorithms.* State spaces  $G$  of nonlinear systems often satisfy geometric constraints: in general, numerical algorithms do not preserve these constraints. We used the Cayley algebra of trees to derive specialized numerical algorithms with the property that by construction they preserve certain geometric structures. It turns out that one of the key technical ideas is to work with the ring  $R$  of all observations  $R = \{f : G \rightarrow \mathbf{R}\}$  of the state space  $G$  rather than with the state space  $G$  itself. This approach allows numerical algorithms to be derived in an invariant or basis independent fashion [Crouch93]. Specific numerical algorithms were derived by using the Cayley algebra of trees and viewing  $R$  as a  $\mathcal{T}$ -module. In particular, we derived application-specific numerical algorithms for a rotating rigid body ( $G = SO(3)$ ) and a ball rolling on an inclined plane

( $G = E(3)$ ) in [Crouch95].

*Hybrid Systems.* Discrete systems, such as automata, and continuous systems, such as control systems, have both internal representations involving states, and external representations involving inputs and outputs. The external representation can be represented by a formal series or generating series  $p$ . Not all such series are associated with systems. A realization theorem gives conditions on  $p$  which describes the system externally so that there exists an internal representation involving states.

We have already mentioned how Hopf algebras arise and can be used to organize combinatorial computations, especially those that arise in symbolic computation. In [Grossman92a], we show how there is a natural Hopf algebra  $H$  associated to a nonlinear system with inputs and outputs and how the formal series  $p$  can be viewed as element of the dual  $H^*$ . The Hopf algebra  $H$  acts on the space of all observations  $R$  of the system.

This led to a new class of realization theorems. In the case of nonlinear systems with input and outputs the realization theorem becomes the Fliess Theorem [Grossman92a]; in the case of finite automata, the realization theorem becomes the Myhill-Nerode theorem [Grossman95a].

This suggested that hybrid systems be defined through an internal description involving observation functions  $R$  rather than through an internal description involving states. From this viewpoint [Grossman95a], a *hybrid system* consists of a commutative algebra  $R$  and a Hopf algebra  $H$ , together with an action of  $H$  on  $R$ . The action expresses the dynamics. The extreme cases of a nonlinear control system and an automaton are recovered by assuming special forms for  $H$ . A hybrid system simply consists of a general  $H$ , which corresponds to a collection of interacting nonlinear control systems and automata. In [Grossman95c] these ideas were used to reveal a structural similarity between hybrid systems and quantum automata.

*TrajectoryStores.* An interesting class of hybrid systems arises by considering a collection of control systems, each one corresponding to a separate mode of the hybrid system. Transitions between modes are specified by an automaton. These are *mode-switched* hybrid systems. Each state of a finite automaton corresponds to a nonlinear control system describing an underlying physical system in a particular regime. Such hybrid systems are a special case of the definition above. An important open problem for such systems is: Given a sequence of points in state space, find a path connecting them using flows of the underlying control systems and satisfying the mode changes allowed by the underlying automaton. This is the path planning

problem.

Trying to compute such paths in real time is usually prohibitively expensive, except in special cases. On the other hand, one can store many *precomputed* trajectory segments and try to select an appropriate sequence of them which approximate the desired path connecting the sequence of given points. This is the approach we took [Grossman93b]. It involves managing large collections of trajectory segments or TrajectoryStores using object managers.

*PTool.* Our solution to path planning for hybrid systems is a variant of the classical approach of trading space for time. In our case, we traded high performance computing for high performance data management. In particular, over several years, we developed a persistent object manager called PTool [Grossman94a], [Grossman95b], and [Grossman95d] which supports the type of high performance data management required in scientific computing. Unlike, a traditional database management system, PTool is optimized for the type of data access for applications such as the TrajectoryStore described above. Recently, we have developed variants of PTool suited for clustered computing environments. In [Grossman94c], we described how PTool could be used for hybrid systems involving rules and agents.

*Other Activities.* We organized a workshop "Application-Specific Symbolic Techniques in High Performance Computing Environments" at the Fields Institute in Waterloo, Ontario from October 18-20, 1993. The workshop was funded by \$7500 provided by NASA grant NAG 2-513 and \$7500 provided by the Fields Institute.

Approximately fifty scientists, engineers, and computer scientists attended the workshop, which focused on three themes:

1. using symbolic methods to derive application specific numerical algorithms
2. using symbolic methods to analyze nonlinear systems
3. algorithms and software for hybrid systems.

### **3 Publications Written During the Report Period**

[Crouch92] P.E. Crouch and R. L. Grossman, "The Explicit Computation of Integration Algorithms and First Integrals for Ordinary Differential Equations With Polynomial Coefficients Using Trees," *Proceedings of*

*the 1992 International Symposium on Symbolic and Algebraic Computation*, ACM Press, pp. 89–94.

- [Crouch93] P. Crouch and R. Grossman, “Numerical integration of ordinary differential equations on manifolds,” *Journal of Nonlinear Science*, Volume 3, pp. 1–33, 1993.
- [Crouch95] P. E. Crouch, R. Grossman, Y. Yan “On the numerical integration of the rolling ball equations using geometrically exact algorithms,” *Mechanics of Structures of Machines*, to appear.
- [Doffou95] M. J. Doffou and R. L. Grossman, “The Symbolic Computation of Differential Invariants of Polynomial Vector Field Systems Using Trees,” *Proceedings of the 1995 International Symposium on Symbolic and Algebraic Computation*, A. H. M. Levelt, editor, ACM, 1995, pp. 26–31.
- [Grossman92a] R. Grossman and R. G. Larson, “The realization of input-output maps using bialgebras,” *Forum Mathematicum*, Volume 4, pp. 109–121, 1992.
- [Grossman92b] R. Grossman and R. Larson, “The symbolic computation of derivations using labeled trees,” *Journal of Symbolic Computation*, Volume 13, pp. 511–523, 1992.
- [Grossman92c] R. L. Grossman, R. G. Larson “Viewing hybrid systems as products of control systems and automata,” *Proceedings of the 31st IEEE Conference on Decision and Control*, IEEE Press, 1992, pp. 2953–2955.
- [Grossman93a] R. L. Grossman and R. L. Larson, “Some Remarks About Flows in Hybrid Systems,” in R. L. Grossman, A. Nerode, A. P. Ravn, and H. Rischel, editors, *Hybrid Systems*, Lecture Notes in Computer Science, Volume 736, Springer-Verlag, New York, 1993, pp. 357–365.
- [Grossman93b] R. L. Grossman, D. Valsamis and X. Qin, “Persistent stores and hybrid systems,” *Proceedings of the 32nd IEEE Conference on Decision and Control*, IEEE Press, 1993, pp. 2298–2302.
- [Grossman93c] R. L. Grossman, A. Nerode, A. P. Ravn, and H. Rischel, editors, *Hybrid Systems*, Lecture Notes in Computer Science, Volume 736, Springer-Verlag, New York, 1993.

- [Grossman94a] R. L. Grossman and X. Qin, "Ptool: a scalable persistent object manager," *Proceedings of SIGMOD 94*, ACM, 1994, page 510.
- [Grossman94b] R. L. Grossman, W. Sluis, and W. Shadwick, "On Nonlinear Normal Forms," *Proceedings of the 33rd IEEE Conference on Decision and Control*, IEEE Press, 1994.
- [Grossman94c] R. L. Grossman, A. Nerode, and W. Kohn, "Nonlinear Systems, Automata, and Agents: Managing their Symbolic Data Using Light Weight Persistent Object Managers," *International Symposium on Fifth Generation Computer Systems, 1994: Workshop on Heterogeneous Cooperative Knowledge-Bases*, Kazumasa Yokota, editor, ICOT, pp. 65-74.
- [Grossman95a] R. L. Grossman and R. G. Larson, "An algebraic approach to hybrid systems," *Journal of Theoretical Computer Science*, Volume 138, pp. 101-112, 1995.
- [Grossman95b] R. L. Grossman, N. Araujo, X. Qin, and W. Xu, "Managing physical folios of objects between nodes," *Proceedings of the Sixth International Workshop on Persistent Object Systems*, Springer-Verlag, 1995, to appear.
- [Grossman95c] R. L. Grossman and M. Sweedler "Hybrid Systems and Quantum Automata: Preliminary Announcement," *Hybrid Systems II*, P. Antsaklis, W. Kohn, A. Nerode, S. Sastry, editors, Springer Lecture Notes in Computer Science, 1995, to appear.
- [Grossman95d] R. L. Grossman, D. Hanley, and X. Qin, "PTool: A Light Weight Persistent Object Manager," *Proceedings of SIGMOD 95*, ACM, 1995, to appear.