The symbolic computation of series solutions to ordinary differential equations using trees (extended abstract)

Robert Grossman*
University of Illinois at Chicago

November, 1991

Abstract

In this note, we explain how the algorithms in [11], [5] and [6] immediately give formulas which can be used for the efficient symbolic computation of series expansions to solutions of nonlinear systems of ordinary differential equations. As a by product of this analysis, we derive formulas relating trees to the coefficients of the series expansions, similar to the work by Leroux and Viennot [15], [16], [17], and Lamnabhi, Leroux and Viennot [18].

1 Introduction

In this note, we explain how the algorithms in [11], [5] and [6] immediately give formulas which can be used for the efficient symbolic computation of series expansions to solutions of nonlinear systems of ordinary differential equations. As a by product of this analysis, we derive formulas relating trees to the coefficients of the series expansions, similar to the work by Leroux and Viennot [15], [16], [17], and Lamnabhi, Leroux and Viennot [18]. In this section and the next, we follow the exposition in [14]. This is an extended abstract: a complete version will be published elsewhere.

We now describe the basic idea of how trees can be used to organize computations involving vector fields following [8] and [7]. For background

1

material on Hopf algebras, see [19]. Consider a control system

$$\dot{x}(t) = E_1(x(t)) + u_1(t)E_2(x(t)) + u_2(t)E_3(x(t)), \quad x(0) = x^0 \in \mathbf{R}^N, \quad (1)$$

where E_1, E_2 and E_3 are vector fields defined in a neighborhood of $x^0 \in \mathbf{R}^N$ and $t \mapsto u_i(t)$ are controls. Our goal is to describe a class of algorithms for the effective symbolic computation of expressions built from the vector fields that describe the local behavior of the control system. These expressions include iterated Lie brackets and the generating series of the system.

The starting point is to assign trees to vector fields as illustrated in Figure 1, and then to impose a multiplication on the trees which is compatible with the composition of vector fields. Assume that the vector fields E_j have the form:

$$E_j = \sum_{\mu=1}^{N} a_j^\mu D_\mu, \quad j = 1,2,3, \quad j = 1,\dots,M, \quad (2)$$

where a_j^μ are smooth functions on $\mathbf{R}^N$ and $D_\mu = \partial/\partial x_\mu$. Now

$$E_2 \cdot E_1 = \sum b_j(D_j a_i)D_i + \sum b_j a_i D_j D_i$$

and $E_3 \cdot E_2 \cdot E_1$ is equal to

$$\sum a_3^{\mu_k}(D_k a_2^{\mu_j})(D_j a_1^{\mu_i})D_i + \sum a_3^{\mu_k} a_2^{\mu_j}(D_k D_j a_1^{\mu_i})D_i + \sum a_3^{\mu_k} a_2^{\mu_j}(D_j a_1^{\mu_i})D_k D_i$$

$$+ \sum a_3^{\mu_k} a_2^{\mu_j} a_1^{\mu_i} D_k D_j D_i + \sum a_3^{\mu_k} a_2^{\mu_j}(D_k a_1^{\mu_i})D_j D_i + \sum a_3^{\mu_k} a_1^{\mu_i}(D_k a_2^{\mu_j})D_j D_i.$$

$$(3)$$

Here the sum is for $i,j,k = 1,\dots,N$ and hence involves $O(N^3)$ differentiations. It is convenient to keep track of the terms that arise in this way using labeled trees: we indicate in Figure 2 the trees that are associated with the six sums in this expression.

An iterated Lie bracket such as

$$[E_3,[E_2,E_1]] = E_3 E_2 E_1 - E_3 E_1 E_2 - E_2 E_1 E_3 + E_1 E_2 E_3 \quad (4)$$

gives rise in this fashion to 24 trees corresponding to the $24N^3$ differentiations that a naive computation of this expression requires. On the other hand, 18 of the trees cancel, saving us from computing $18N^3$ terms. We are left with $6N^3$ terms of the form $(junk)D_{\mu_1}$. A careful examination of this correspondence between labeled trees and expressions involving the E_j's shows that the composition of the vector fields E_j's, viewed as first order differential operators, corresponds to a multiplication on trees. This multiplication is illustrated in Figure 3. It turns out that this construction yields an algebra, which we call the *algebra of Cayley trees.*

2 The algebra of Cayley trees

In this section, we follow [13] and define a bialgebra structure on spaces of trees. The relation between trees and differential operators goes back at least as far as Cayley [3] and [4]. Of this literature, the work most closely related to the view point taken here is Butcher's use of trees to analyze Runge-Kutta algorithms [1] and [2].

Let k will denote a field of characteristic 0. By a *tree* we will mean a finite rooted tree. Let T be the set of finite rooted trees, and let $k\{T\}$ be the k-vector space which has T as a basis.

We first define the coalgebra structure on $k\{T\}$. If $t \in T$ is a tree whose root has children $s_1, \ldots, s_r$, the coproduct $\Delta(t)$ is the sum of the 2^r terms $t_1 \otimes t_2$, where the children of the root of t_1 and the children of the root of t_2 range over all 2^r possible partitions of the children of the root of t into two subsets. The augmentation ϵ which sends the trivial tree to 1 and every other tree to 0 is a counit for this coproduct. It is immediate that comultiplication is cocommutative.

We next define the algebra structure on $k\{T\}$. Suppose that $t_1, t_2 \in T$ are trees. Let $s_1, \ldots, s_r$ be the children of the root of t_1. If t_2 has $n + 1$ nodes (counting the root), there are $(n+1)^r$ ways to attach the r subtrees of t_1 which have $s_1, \ldots, s_r$ as roots to the tree t_2 by making each s_i the child of some node of t_2. The product $t_1 t_2$ is defined to be the sum of these $(n+1)^r$ trees. It can be shown that this product is associative, and that the trivial tree consisting only of the root is a right and left unit for this product. It can also be shown that the maps defining the coalgebra structure are algebra homomorphisms, so that $k\{T\}$ is a bialgebra. For details, see [9].

The bialgebra $k\{T\}$ is graded: $k\{T\}_n$ has as basis all trees with $n + 1$ nodes. Because the bialgebra $k\{T\}$ is graded connected, it is a Hopf algebra. We summarize the above discussion in the following theorem.

Theorem 2.1 *The vector space $k\{T\}$ with basis the set of finite rooted trees is a cocommutative graded connected Hopf algebra.*

Assume now that each node of the tree (except for the root) is labeled with a symbol from the set $\{E_1, \ldots, E_M\}$. We can define the product and coproduct as above, and, once again, the resulting space is a bialgebra. See [9] for details. Let $k\{\mathcal{L}T\}$ denote this algebra.

Let R denote a subring of the commutative ring of smooth functions on $\mathbf{R}^N$. We now define an action of the algebra of Cayley trees

$$B = k\{\mathcal{L}T\}$$

3

on the ring R, making R a B-module algebra, which captures the action of trees as higher derivations. This requires that we interpret the formal symbols E_j as derivations of R using Equations 2. The action is defined using the map

$$\psi : k\{\mathcal{LT}\} \to \mathrm{End}_k\, R,$$

as follows:

1. Given a labeled, ordered tree t with $m + 1$ nodes, assign the root the number 0 and assign the remaining nodes the numbers $1, \ldots, m$. We identify the node with the number assigned to it. To the node k associate the summation index μ_k. Denote $(\mu_1, \ldots, \mu_m)$ by μ.

2. For the labeled tree t, let k be a node of t, labeled with E_{γ_k} if $k > 0$, and let $l, \ldots, l'$ be the children of k. Define

$$
\begin{aligned}
R(k;\mu) &= D_{\mu_l} \cdots D_{\mu_{l'}} a_{\gamma_k}^{\mu_k}, \quad \text{if } k > 0 \text{ is not the root;} \\
&= D_{\mu_l} \cdots D_{\mu_{l'}}, \quad \text{if } k = 0 \text{ is the root.}
\end{aligned}
$$

Note that if $k > 0$, then $R(k;\mu) \in R$.

3. Define

$$\psi(t) = \sum_{\mu_1, \ldots, \mu_m = 1}^{N} R(m;\mu) \cdots R(1;\mu) c(0;\mu).$$

4. Extend ψ to all of $k\{\mathcal{LT}\}$ by linearity.

It is straightforward to check that this action of B on R makes R into a B-module algebra.

We summarize with the following theorem.

Theorem 2.2 *Let R the algebra of smooth functions on $\mathbf{R}^N$. Let B denote the algebra of Cayley trees $k\{\mathcal{LT}\}$. Then R is a B-module algebra with respect to the action defined by ψ.*

3 Taylor flows for vector fields with polynomial coefficients

The standard action of the algebra A of differential operators generated by $E_1, \ldots, E_M$ on the algebra of smooth functions R gives R the structure of a A-module algebra. It is easy to relate this H-module algebra structures on R to the H-module algebra structure defined in the section above.

4

Let

$$\phi : A \longrightarrow B$$

denote the map sending the generator E_j of the algebra A to the tree consisting of two nodes: the root and a single child labeled E_j. This map is illustrated in Figure 1. Extend ϕ to be an algebra homomorphism. Let χ denote the map

$$A \to \mathrm{End}_k R$$

defined by using the substitution (2) and simplifying to obtain an endomorphim of R. We have the following diagram:

$$
\begin{array}{ccc}
A & \to & B \\
& \searrow & \downarrow \\
& & \mathrm{End}_k R
\end{array}
\tag{5}
$$

Theorem 3.1 *(i) The maps χ, ϕ and ψ are related by $\chi = \psi \circ \phi$. (ii) Fix a function $f \in R$ and a differential operator $p \in A$. Then*

$$p \cdot f = \phi(p) \cdot f.$$

Here the action on the left views R as an A-module algebra, while the action on the right views R as B-module algebra.

The first assertion is proved in [12] and the second assertion follows from the first assertion and the definitions.

To simplify the notation, in the remainder of this section we restrict attention to a single derivation F. We will need the following Theorem.

Theorem 3.2 *Assume $f \in R$ and $F \in \mathrm{Der}(R)$. Making use of the actions defined in Theorem 3.1, we have:*
(i) For $k \geq 0$,

$$(F^k f)(x) = \frac{d^k}{dt^k} f(\exp(tF)x) \mid_{t=0}.$$

(ii) If f is analytic near x, then for sufficiently small t,

$$f(\exp(tF)x) = \sum_{k=0}^{\infty} f(x; F^k) \frac{t^k}{k!},$$

where $f(x; F^k)$ is defined to be $(F^k f)(x)$.
(iii) If f is analytic near x, then for sufficiently small t,

$$f(\exp(tF)x) = \exp(t\phi(F)) \cdot f|_x.$$

PROOF. Assertions (i) and (ii) can be found in [20]. Theorem 3.1 and Assertion (ii) then give Assertion (iii).

Assume now that the derivation F is of the form

$$F = \sum_{\mu_1=1}^{N} b^{\mu_1} D_{\mu_1} + \sum_{\mu_1,\mu_2=1}^{N} b^{\mu_1}_{\mu_2} x^{\mu_2} D_{\mu_1}$$

$$+ \sum_{\mu_1,\mu_2,\mu_3=1}^{N} b^{\mu_1}_{\mu_2,\mu_3} x^{\mu_3} x^{\mu_2} D_{\mu_1} + \cdots$$

where all except a finite number of the coefficients $b^{\mu_1}_{\mu_2,\ldots,\mu_k} \in k$ are zero.

We let $kT\{F,x\}$ denote the vector space with basis the set of finite, rooted trees, whose nodes are labeled with the symbols F and x, and let $k<F>$ denote the free associative algebra generated by the symbol F. We define

$$\phi : k<F> \longrightarrow kT\{F,x\}$$

on the generator F as indicated in Figure 4, and extend ϕ to all of $k<F>$ by making it an algebra homomorphism. Figure 5 illustrates the multiplication in $kT\{F,x\}$.

We define a map

$$\psi : kT\{F,x\} \longrightarrow \mathrm{Diff}(R)$$

as follows:

1. Given a tree $t \in kT\{F,x\}$ with $m + 1$ nodes, assign the root the number 0 and assign the remaining nodes the numbers $1, \ldots, m$.

2. Consider a node j, with children $k, \ldots, k'$ labeled with an F and children $l, \ldots, l'$ labeled with an x. Define

$$R(j;\mu) = D_{\mu_k} \cdots D_{\mu_{k'}} b^{\mu_j}_{\mu_k,\ldots,\mu_{k'},\mu_l,\ldots,\mu_{l'}},$$

if the node is labeled with a F;

$$= D_{\mu_k} \cdots D_{\mu_{k'}} x^{\mu_j},$$

if the node is labeled with a x;

$$= D_{\mu_k} \cdots D_{\mu_{k'}},$$

if the node is the root

(Note that in the first case, $R(j;\mu)$ is zero, since the b's are constant.)

3. Define

$$\psi(t) = \sum_{\mu_1,\dots,\mu_m=1}^{N} R(m;\mu)\cdots R(1;\mu)R(0;\mu).$$

4. Extend ψ to all of $kT\{F,x\}$ by linearity.

We can now state our main theorem.

Theorem 3.3 *Let the derivation F be of the form*

$$F = \sum_{\mu_1=1}^{N} b^{\mu_1} D_{\mu_1} + \sum_{\mu_1,\mu_2=1}^{N} b^{\mu_1}_{\mu_2} x^{\mu_2} D_{\mu_1}$$

$$+ \sum_{\mu_1,\mu_2,\mu_3=1}^{N} b^{\mu_1}_{\mu_2,\mu_3} x^{\mu_3} x^{\mu_2} D_{\mu_1} + \cdots,$$

where $b^{\mu_1}_{\mu_2,\dots,\mu_k} \in k$ and at most a finite number are nonzero. Then the ith component of the nth term in the formal series expansion of the flow

$$\dot{x}(t) = F(x(t)), \quad x(0) = x^0, \quad x(t) \in \mathbf{R}^N$$

is given by

$$\left(\sum_u u\right) \cdot x^i|_{x=x^0},$$

where the sum is over all labeled trees $u \in kT\{F,x\}$ occuring in the product $(\phi(F))^n$. Here the action of a tree u on the function x^i is determined by viewing R as a B-module algebra using the action just defined above, and $\phi(F)$ is defined as in Figure 4.

This follows from Theorems 3.1 and 3.2.

References

[1] J. C. Butcher, "An order bound for Runge-Kutta methods," *SIAM J. Numerical Analysis*, **12** (1975), pp. 304–315.

[2] J. C. Butcher, *The Numerical Analysis of Ordinary Differential Equations*, John Wiley, 1986.

[3] A. Cayley, "On the theory of the analytical forms called trees," in *Collected Mathematical Papers of Arthur Cayley*, Cambridge Univ. Press, Cambridge, 1890, Vol. 3, pp. 242–246.

[4] A. Cayley, "On the analytical forms called trees. Second part," in *Collected Mathematical Papers of Arthur Cayley*, Cambridge Univ. Press, Cambridge, 1891, Vol. 4, pp. 112–115.

[5] P. Crouch, R. Grossman, and R. G. Larson, "Trees, bialgebras, and intrinsic numerical algorithms," *Laboratory for Advanced Computing Technical Report* Number LAC90-R23, University of Illinois at Chicago, May, 1990.

[6] P. Crouch, R. Grossman, and R. G. Larson, "Computations involving differential operators and their actions on functions," *Proceedings of 1991 International Symposium on Symbolic and Algebraic Computation*, ACM, 1991, pp. 301–307.

[7] R. Grossman, "Using trees to compute approximate solutions of ordinary differential equations exactly," *Computer Algebra and Differential Equations*, M. F. Singer, editor, Academic Press, New York, 1991, in press.

[8] R. Grossman, "The evaluation of expresssions involving higher order derivations," *Journal of Mathematical Systems, Estimation, and Control*, Vol. 1, 1990.

[9] R. Grossman and R. Larson, "Hopf algebraic structures of families of trees," *J. Algebra*, Vol. 26 (1989), pp. 184–210.

[10] R. Grossman and R. Larson, "Labeled trees and the efficient computation of derivations," in *Proceedings of 1989 International Symposium on Symbolic and Algebraic Computation*, ACM, 1989, pp. 74-80.

[11] R. Grossman and R. Larson, "Solving nonlinear equations from higher order derivations in linear stages," *Advances in Mathematics*, Vol. 82, pp. 180–202, 1990.

[12] R. Grossman and R. Larson, "The symbolic computation of derivations using labeled trees," *Journal of Symbolic Computation*, to appear.

[13] R. Grossman and R. G. Larson, "Hopf-algebraic structure of combinatorial objects and differential operators," *Israeli J. Math.*, Vol. 72, 1990, pp. 109–117.

[14] R. Grossman and R. G. Larson, "The symbolic computation of vector field expressions," *Proceedings of Algebraic Computing in Control*

91, edited by G. Jacob and F. Lamnabhi-Lagarrigue, Springer-Verlag, Berlin, in press.

[15] P. Leroux and X. G. Viennot, "Combinatorial resolution of systems of differential equations, I: Ordinary differential equations," in *Combinatoire Énumérative, UQAM 1985, Proceedings, Lecture Notes in Mathematics,* Volume 1234, Springer-Verlag, 1986, pp. 210–245.

[16] P. Leroux and X. G. Viennot, "Combinatorial systmes of differential equations, IV: Separation of variables," in *Proceedings of First Japan Conference on Graph Theory and Applications, June, 1986, Discrete Mathematics,* Volume 72, pp. 237–250, 1988.

[17] P. Leroux and X. G. Viennot, "A combinatorial approach to nonlinear functional expansions," *27th Conference on Decision and Control,* Austin, Texas, pp. 1314–1319, 1988.

[18] F. Lamnabhi-Lagarrigue, P. Leroux and X. G. Viennot, "Combinatorial interpretation of Brockett bilinear approximations," to appear.

[19] M. E. Sweedler, *Hopf Algebras,* Benjamin, 1969.

[20] V. S. Varadarajan, "Lie Groups, Lie Algebras, and their Representations," Prentice-Hall, Inc., 1974.

$E1 \longmapsto \overset{\circ}{\underset{\circ}{|}}_{E1}$, $E2 \longmapsto \overset{\circ}{\underset{\circ}{|}}_{E2}$, $E3 \longmapsto \overset{\circ}{\underset{\circ}{|}}_{E3}$
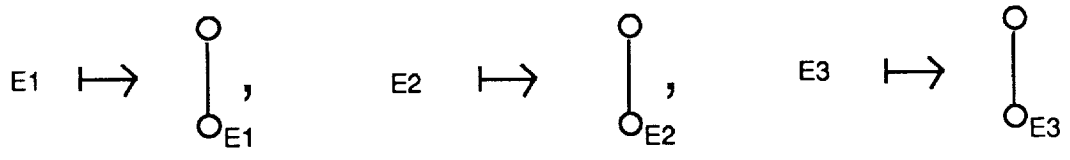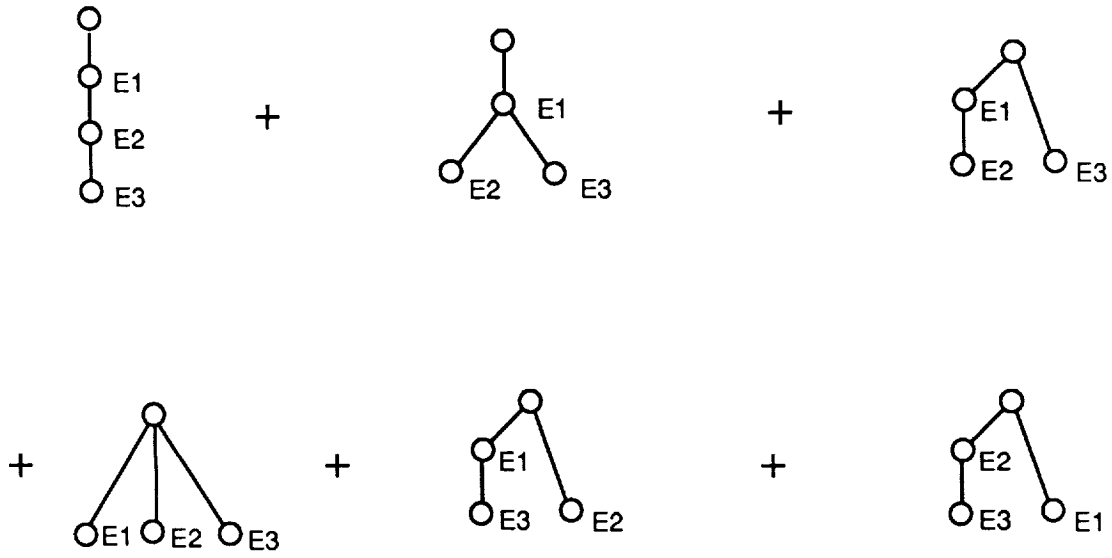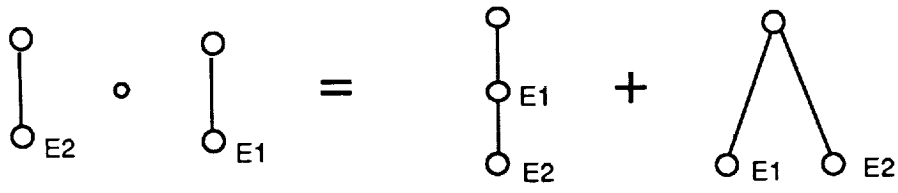
Figure 1

Figure 2

Figure 3

Figure 4

Figure 5