J. Symbolic Computation (1985) 1, 101–107

Computer Algebra Application for Classification of Integrable Non-linear Evolution Equations

V. P. GERDT, A. B. SHVACHKA AND A. YU. ZHARKOV

Laboratory of Computing Techniques and Automation, Joint Institute for Nuclear Research, Head Post Office, P.O. Box 79, 101000 Moscow, USSR

(*Received* 2 *November* 1984)

The application of computer algebra for classification of integrable non-linear evolution equations is discussed. Algorithms for testing conditions of formal integrability, to calculate the Lie–Bäcklund symmetries and conservation law densities are developed and implemented on the basis of the computer algebra system PL/1–FORMAC.

1. Introduction

In recent years a good deal of attention has been paid to the classification of integrable non-linear evolution equations

$$\begin{gathered} u_t = F(u, u_1, \ldots, u_n), \\ u \equiv u(x, t), \qquad u_i \equiv \frac{d^i u}{dx^i}. \end{gathered} \tag{1}$$

Different criteria of integrability are used for the classification of equations (1): the existence of non-trivial symmetries (Fokas, 1980; Fujimoto & Watanabe, 1983), conservation laws (Abellanas & Galindo, 1983), prolongation structures (Leo *et al.*, 1983). In this paper we shall describe a classification method based on the concept of formal integrability (Ibragimov & Shabat, 1980b). The latter is one of the strict formulations of the concept of L–A pair (Lax, 1968). Formal integrability puts strict limitations on the form of the right-hand side of (1) and allows us to find all the evolution equations possessing a non-trivial Lie–Bäcklund algebra and infinite series of non-trivial conservation laws. Among them there are equations interesting from the physical point of view because they have soliton solutions, e.g. the *KdV* equations.

Recently the problem of classification of formally integrable evolution equations has been solved for the third order equations of the form $u_t = u_3 + f(u, u_1, u_2)$ (Svinolupov & Sokolov, 1982). The classification of higher order equations demands tedious computations. To carry them out automatically we developed the program FORMINT based on the computer algebra system PL/1–FORMAC (Bahr, 1973). The program allows one to check the conditions of formal integrability for F, to obtain the equivalent equations to those on the F function, to find the non-trivial elements of the Lie–Bäcklund algebra (symmetries), and to compute the conservation law densities.

In the second section of this paper the basic concepts and results of the theory of formally integrable systems are given. In the third section the structure of the algorithms solving the above problems are briefly described. In the final section examples of using the

0747–7171/85/010101+07 \$03.00/0

program are given, some concrete equations, of particular interest in mathematical physics, are treated, and the computational experience in the FORMAC as well as the REDUCE-2 environment is discussed.

2. Mathematical Background

Equation (1) is called formally integrable if there is a formal series

$$L = \sum_{i=-\infty}^{1} a_i D^i$$
$$a_i \equiv a_i(u, u_1, \ldots, u_{k_i}), \qquad D \equiv \frac{\mathrm{d}}{\mathrm{d}x}, \qquad D^{-1}D = DD^{-1} = 1, \tag{2}$$

satisfying the operator relation

$$L_t - [F_*, L] = 0, \tag{3}$$

where

$$F_* \equiv \sum_{i=0}^{n} \frac{\partial F}{\partial u_i} D^i, \qquad L_t \equiv \left[\frac{\mathrm{d}}{\mathrm{d}t}, L\right], \qquad \frac{\mathrm{d}}{\mathrm{d}t} \equiv \sum_i (D^i F) \frac{\partial}{\partial u_i}.$$

A conservation law density for the equation (1) is a function $P(u, u_1, \ldots, u_k)$ such that

$$\frac{\mathrm{d}}{\mathrm{d}t} P \in ImD. \tag{4}$$

The notation $\in ImD$ means that left-hand side expression is a gradient (i.e. total derivative with respect to x). If $P \in ImD$ then the conservation law is called trivial.

The Lie–Bäcklund algebra for (1) is the set of functions $H(u, u_1, \ldots, u_m)$ such that

$$H_* F - F_* H = 0. \tag{5}$$

The algebra is called non-trivial if it contains elements (symmetries) different from u_1 and F.

The following theorems establish the connections between the concepts introduced above.

THEOREM 1. *If equation* (1) *has an infinite Lie–Bäcklund algebra, then it is formally integrable (Ibragimov & Shabat,* 1980*b).*

THEOREM 2. *If equation* (1) *has an infinite sequence of non-trivial conservation laws, then it is formally integrable (Svinolupov & Sokolov,* 1982).

THEOREM 3. *The formal integrability of* (1) *is equivalent to the property that there is an infinite sequence of conservation laws of the following type (Ibragimov & Shabat,* 1980*b)*

$$\frac{\mathrm{d}}{\mathrm{d}t} R_i \in ImD, \quad i = -1, 0, 1, 2, 3, \ldots, \tag{6}$$

where

$$R_{-1} = \mathrm{Res}\,(L^{-1}), \qquad R_0 = \mathrm{Res}\,(L^{-1}L_t),$$
$$R_m = \mathrm{Res}\,(L^m), \qquad m = 1, 2, 3, \ldots, \tag{7}$$

and

$$Res\,(\textstyle\sum a_i D^i) \equiv a_{-1}.$$

3. Algorithms

Theorem 3 allows one to solve the classification problem for formally integrable evolution equations of order n up to transformations of the form $u(x, t) = \phi(v(x, t))$. Our algorithmic solution is then:

(1) Derivation of all the F, for which the first conditions (6) hold (e.g. $i = -1, 0, 1, 2, 3, 4$; primary classification).
(2) Checking the higher conditions (6) for the equations derived.
(3) Testing the equations obtained for the property of having non-trivial symmetries.
(4) Derivation of the Bäcklund transformations that connect different equations of the list obtained (using the well-known techniques of Svinolupov *et al.* (1983)).

We now give more details for (1)–(4).

3.1 Testing the conditions of formal integrability

To test conditions (6) for the given equation (1) it is necessary primarily to express the densities R_i in terms of the given functions $F(u, u_1, \ldots, u_n)$. It can be easily shown that

$$R_{-1} = (F_n)^{-1/n}, \qquad R_0 = F_{n-1}/F_n, \tag{8}$$

where

$$F_i \equiv \frac{\partial F}{\partial u_i}$$

and the densities R_m can be derived from the following recurrence relations on the coefficients of the series $L^m = \sum_{i=-\infty}^{m} a_i D^i$

$$a_m = (F_n)^{m/n}, \quad m = 1, 2, 3, \ldots,$$
$$a_i = \frac{(F_n)^{i/n}}{n} D^{-1}\left[(F_n)^{-\frac{(i+n)}{n}} \cdot G_{n+i-1}|_{a_i=0} + \frac{\mathrm{d}}{\mathrm{d}t} a_{n+i-1}\right], \quad i = m-1, m-2, m-3, \ldots, \tag{9}$$

where C_{n+i-1} are the coefficients of D^{n+i-1} in the commutator $[L^m, F_*]$, $a_{n+i-1} \equiv 0$ if $i > m-n+1$, and the integration constants in (9) are zeros. The recurrence relations (9) may be obtained from (3), which is valid not only for L but also for L^m.

Then one has to check that $S \equiv \mathrm{d}R_i/\mathrm{d}t \in ImD$. The corresponding algorithm based on the linear dependence of the gradient $S(u, u_1, \ldots, u_k)$ on the highest derivative u_k is given below.

(1) $S := \dfrac{\mathrm{d}R_i}{\mathrm{d}t}$; (10)

(2) $k :=$ highest order of derivatives u_k occurring in S;

(3) if $\partial^2 S/\partial u_k^2 \neq 0$ then STOP (check of the linearity condition);

(4) $S := S - D \displaystyle\int \frac{\partial S}{\mathrm{d}u_k} \cdot \mathrm{d}u_{k-1}$ (reducing the order of S);

(5) if $S \neq 0$ then go to (2) else STOP.

If the S expressions contain parameters or undetermined functions, then algorithm (10) does not stop at step 3 but continues after setting the terms in S non-linearly depending on u_k equal to zero and obtaining the corresponding equations for the parameters and/or the undetermined functions.

3.2 PRIMARY CLASSIFICATION

The primary classification of the evolution equations (1) is done according to the scheme described in 3.1 where the form of $F(u, u_1, \ldots, u_n)$ is completely or partially unknown. One can obtain differential equations on F according to (8), (9), (10). The solution of these equations gives the possible forms of F. The most tedious part of the computation (derivation of the equations on F) can be carried out totally automatically by FORMINT (see example 3, section 4). The solution of these equations may sometimes only be possible by routines from computer algebra systems, depending on the form of equations. (For example, if only unknown numerical parameters occur in F, then the equations are algebraic and may be solved by the routines described in Buchberger (1985).)

The evolution equations with the functions F obtained by the above procedures for small indices i (e.g. $i = -1, 0, \ldots, 4$, see (6)) are then checked using the same procedure for higher indices i. As a result, part of the evolution equations is rejected.

3.3 SYMMETRIES

The algorithm for finding the symmetry $H(u, u_1, \ldots, u_m)$ of the given order m for a given equation (1) is based on the following relations (Ibragimov & Shabat, 1980b)

$$\frac{\partial H}{\partial u_i} = a_i, \quad i = 2, \ 3, \ \ldots, m, \tag{11}$$

where a_i are the coefficients of the series L^m. One can find $H(u, u_1, \ldots, u_m)$ up to addition of an arbitrary function $h(u, u_1)$ by expressing a_i through F according to (9), checking the compatibility conditions for the system (11)

$$\frac{\partial a_i}{\partial u_j} - \frac{\partial a_j}{\partial u_i} = 0, \quad i \neq j \tag{12}$$

and, in case (12) holds, by integration of (11). After that one must substitute the result into (5), obtain equations on h, and solve them. In the important special case, when $F = u_n + f(u, u_1, \ldots, u_{n-2})$, equations (11) are valid for $i = 0, 1, 2, \ldots, m$ and H can be obtained by simple integration of (11).

The algorithm described is completely implemented in FORMINT except solving the equations on $h(u, u_1)$. Note that, according to theorem 1, the non-trivial symmetry property of the evolution equation considered is a strong argument for its formal integrability since there is no example of an evolution equation with non-trivial but finite Lie–Bäcklund algebra.

3.4 BÄCKLUND TRANSFORMATIONS

A Bäcklund transformation from the evolution equation

$$v_t = G(v, v_1, \ldots, v_n) \tag{13}$$

to equation (1) is a transformation

$$v = \phi\,(u, u_1, \ldots, u_m), \quad m \geqslant 1 \tag{14}$$

for which the relation

$$\phi_* F = G(\phi, D\phi, \ldots, D^n\phi) \tag{15}$$

holds. The algorithm for constructing Bäcklund transformations is based on the following fact (Svinolupov *et al.*, 1983): let $R_i(u, u_1, \ldots, u_k)$, $r_i(v, v_1, \ldots, v_l)$, $l \geqslant 1$ be conservation law densities for the equations (1), (13) respectively and let them be non-linear functions of u_k and v_l. (One can easily represent them in such a way by adding suitable gradients.) In the case of a transformation (14) the following relations hold

$$m = k - 1 \tag{16}$$

$$r_i(\phi, D\phi, \ldots, D^l\phi) - R_i(u, u_1, \ldots, u_k) \in ImD. \tag{17}$$

One can find the order m of transformation (14) (or the non-existence of a transformation for a given m) by computing several densities R_i, r_i by (8), (9) and by obtaining equations applying algorithm (10) to the left-hand side of (17). Thereby it is natural to use the simplest R_i, r_i.

It should be mentioned (see Ibragimov & Shabat (1980a)) that the formal integrability of the evolution equation derived from a finite number of conditions (6) is proved by the existence of a Bäcklund transformation of these equations to the known formally integrable equations, e.g. the KdV equation or its higher analogues. For those equations non-transformable to already known equations, the strict proof of their formal integrability presents a theoretically open problem.

4. Computational Examples and Experience

For a given concrete input expression F and a number i subroutine CONDS of FORMINT will produce R_i according to (7) and check whether (6) is satisfied, i.e. whether R_i is a conservation law density.

For a given concrete input expression F and a number j subroutine SYMMTR will produce the symmetry of order j satisfying (5).

For an input expression F with parameters and/or undetermined functions and a number i of conservation law density R_i subroutine INTX yields the equations for the parameters and/or the undetermined functions following from step (3) in algorithm (10).

As a concrete example, for input

$$F := u_5 + u . u_1$$

CONDS yielded

$$\begin{aligned} R_{-1} &= 1, \\ R_0 &= R_1 = R_2 = 0, \\ R_3 &= 3/5u, \\ R_4 &= R_5 = R_6 = 0, \\ R_7 &= (21/50)u^2, \\ R_8 &= 0, \\ R_9 &= (9/25) . u_1^2. \end{aligned}$$

For R_9, CONDS automatically detects that it does not satisfy (6), i.e. R_9 is not a conservation law density in contradiction to the hypothesis expressed in Abellanas & Galindo (1983).

As another example, for the famous Calogero–Degasperis–Fokas equation (Fokas, 1980; Calogero & Degasperis, 1981) as an input

$$F := u_3 - (1/8)u_1^3 + (ae^u + be^{-u})u_1$$

SYMMETR yielded

$$\begin{aligned} H_5 = {} & u_5 + (5/3)abu_1 + (10/3)au_2u_1e^u + (5/3)au_3e^u \\ & + (5/8)u_1^3e^u - (10/3)bu_2u_1e^{-u} + (5/3)bu_3e^{-u} \\ & + (5/8)bu_1^3e^{-u} - (5/8)u_3u_1^2 + (5/6)a^2u_1e^{2u} + (5/6)b^2u_1e^{-2u} \\ & - (5/8)u_2^2u_1 + (3/128)u_1^5. \end{aligned}$$

Finally, for the input

$$F := F(u, u_1, u_2, u_3)$$

and

$$R_{-1} := (F_3)^{-1/3}, \quad F_3 \equiv \partial F/\partial u_3$$

according to (8) INTX gives the ordinary differential equation

$$9F_{3333}(F_3)^2 - 45F_3F_{33}F_{333} + 40(F_{33})^3 = 0.$$

The solution of this equation can be found in Kamke (1969)

$$F_3 = (pu_3^2 + qu_3 + r)^{-3/2},$$

where p, q, r are arbitrary functions of u, u_1, u_2.

The algorithm package described in this paper was implemented, first, in REDUCE-2 (Zharkov & Shvachka, 1983) and, then, also in PL/1-FORMAC (Gerdt *et al.*, 1985). The implementation in REDUCE was easier because REDUCE possesses a comfortable integration package, whereas FORMAC does not have integration facilities at all. However, the REDUCE implementation was too slow and took too much memory space. More importantly, the seventh order symmetry of some equations of fifth order took approximately 1 hour CPU time and 1 Mb memory in the REDUCE implementation and only 3 minutes and 0·3 Mb in the FORMAC implementation (both implementation on the ES 1060 computer at the JINR with an instruction time of approximately 1 μsec). Therefore implementing our own integration subroutine in FORMAC was worthwhile for a limited class of integrands that includes, for example, expressions of the form

$$\int \sum_i (a_i z + b_i)^{c_i} \, dz,$$

where a_i, b_i, c_i are constants.

Readers interested in the details of the programs are referred to Gerdt *et al.* (1985). Also, it is possible to obtain a tape with the program from CPC Program Library, Queen's University of Belfast, N. Ireland.

PROGRAM SUMMARY

Title of program: FORMINT.
Computer: IBM 360/370.
Operating system: OS.
Programming language used: PL/1-FORMAC.
High speed storage required: depends on the problem, minimum 160 000 bytes.
No. of bits in a word: 32.
No. of lines in combined program and test deck: 344.

We are grateful to Profs N. N. Govorun and B. Buchberger for their interest in this work and useful discussions, and to S. I. Svinolupov for consultations.

References

Abellanas, L. A., Galindo, A. (1983). Evolution equations with high-order conservation laws. *J. Math. Phys.* **24**, 504.

Bahr, K. A. (1973). *FORMAC 73 User's Manual.* Darmstadt: GMD/IFV.

Buchberger, B. (1985). Gröbner bases: an algorithmic method in polynomial ideal theory. In: *Recent Trends in Multidimensional Systems Theory* (Bose, N. K., ed.). Dordrecht: Reidel.

Calogero, F., Degasperis, A. (1981). Reduction technique for matrix nonlinear evolution equations solvable by the spectral transform. *J. Math. Phys.* **22**, 23.

Fokas, A. S. (1980). A symmetry approach to exactly solvable evolution equations. *J. Math. Phys.* **21**, 1318.

Fujimoto, A., Watanabe, Y. (1983). Classification of fifth-order evolution equations with nontrivial symmetries. *Math. Jap.* **28**, 43.

Gerdt, V. P., Shvachka, A. B., Zharkov, A. Yu. (1985). FORMINT—a program for classification of integrable nonlinear evolution equations. *Comput. Phys. Commun.* **34**, 303.

Ibragimov, N. H., Shabat, A. B. (1980a). On the Bäcklund transformations for integrable evolution equations. *Funct. Anal.* **14**, 25 (in Russian).

Ibragimov, N. H., Shabat, A. B. (1980b). On the infinite Lie–Bäcklund algebras. *Funct. Anal.* **14**, 79 (in Russian).

Kamke, E. (1969). *Differentialgleichungen: Lösungsmethoden und Lösungen, I* (6th edition). Gewöhnliche Differentialgleichungen, Akad. Leipzig: Verl. Ges. Geest & Portig, K. G.

Lax, P. D. (1968). Integrals of nonlinear equations of evolution and solitary waves. *Comm. Pure Appl. Math.* **21**, 467.

Leo, M., Leo, R. A., Soliani, G., Solombrino, L., Martina, L. (1983). Nonlinear evolution equations and nonabelian prolongations. *J. Math. Phys.* **24**, 1720.

Svinolupov, S. I., Sokolov, V. V. (1982). On the evolution equations with nontrivial conservation laws. *Funct. Anal.*, **16**, 86 (in Russian).

Svinolupov, S. I., Sokolov, V. V., Yamilov, R. I. (1983). On the Bäcklund transformations for integrable evolution equations. *DAN SSSR*, **271**, 802 (in Russian).

Zharkov, A. Yu., Shvachka, A. B. (1983). REDUCE-2 program for investigating integrability of nonlinear evolution equations. *JINR*, R11–83–914, Dubna (in Russian).