A Maple Package for Computing Gröbner Bases for Linear Recurrence Relations

Vladimir P. Gerdt[a][*] and Daniel Robertz[b][†]

[a]Laboratory of Information Technologies, Joint Institute for Nuclear Research, 141980 Dubna, Russia

[b]Lehrstuhl B für Mathematik, RWTH Aachen, Templergraben 64, D-52062 Aachen, Germany

A Maple package for computing Gröbner bases of linear difference ideals is described. The underlying algorithm is based on Janet and Janet-like monomial divisions associated with finite difference operators. The package can be used, for example, for automatic generation of difference schemes for linear partial differential equations and for reduction of multiloop Feynman integrals. These two possible applications are illustrated by simple examples of the Laplace equation and a one-loop scalar integral of propagator type.

1. INTRODUCTION

As it is argued in [1], Gröbner bases form the most universal algorithmic tool for reduction of loop integrals determined by linear recurrence relations derived from the integration by parts method. These recurrence relations can be considered as generators of an ideal in the ring of finite difference polynomials. Hence, to determine the minimal set of basis (master) integrals and to express other integrals in terms of basis integrals it suffices to compute any Gröbner basis, i.e., for any ranking (ordering) on difference operators.

Another important application of difference Gröbner bases is generation of difference schemes for partial differential equations [2,3]. A difference scheme is constructed by elimination of partial derivatives from the system of difference equations obtained by discretization of the differential equations and certain integral relations. Therefore, the corresponding elimination ranking is to be used.

In this paper we present a Maple package for computing Gröbner bases of linear difference ideals. The package is a modified version of our earlier package [4] oriented towards commutative and linear differential algebra and based on the involutive basis algorithm [5]. The modified version is specialized to linear difference ideals

and uses both Janet and Janet-like divisions [6] adopted to linear difference polynomials [7].

2. THE MAPLE PACKAGE LDA

2.1. Outline of the package

The package LDA (abbreviation for **L**inear **D**ifference **A**lgebra) implements the involutive basis algorithm [5][3] for linear systems of difference equations using Janet division. In addition, the package implements a modification of the algorithm oriented towards Janet-like division [6] and, thus, computes Janet-like Gröbner bases of linear difference ideals [7].

Table 1 collects the most important commands of LDA. Its main procedure `JanetBasis` converts a given set of difference polynomials into its Janet basis or Janet-like Gröbner basis form. More precisely, let $\mathbb{R}$ be the difference ring [7] of polynomials in the variables $\theta^\mu \circ y^k$, $\mu \in \mathbb{Z}_{\geq 0}^n$, $k = 1, \ldots, m$, with coefficients in a difference field $\mathbb{K}$ containing $\mathbb{Q}$ for which the field operations can be carried out constructively in Maple. We denote by $\mathbb{R}_L$ the set of linear polynomials in $\mathbb{R}$. Given a generating set $F \subset \mathbb{R}_L$ for a linear difference ideal I in $\mathbb{R}$, `JanetBasis` computes the minimal Janet(-like Gröbner) basis J of I w.r.t. a certain monomial order (ranking). The input for `JanetBasis`

---

[3]However, Janet trees for the fast search of involutive divisors are not yet implemented since Maple does not provide efficient data structures for trees.

consists of the left hand sides of a linear system of difference equations in the dependent variables y^1, ..., y^m, e.g. functions of x_1, ..., x_n. The difference ring $\mathbb{R}$ is specified by the lists of independent variables x_1, ..., x_n and dependent variables given to `JanetBasis`. The output is a list containing the Janet(-like Gröbner) basis J and the lists of independent and dependent variables.

After J is computed, the involutive/$\mathcal{J}-$normal form of any element of $\mathbb{R}_L$ modulo J can be computed using `InvReduce`. Given $p \in \mathbb{R}_L$ representing a residue class $\overline{p}$ of the difference residue class ring $\mathbb{R}/I$, `InvReduce` returns the unique representative $q \in \mathbb{R}_L$ of $\overline{p}$ which is not involutively/$\mathcal{J}-$reducible modulo J. A $\mathbb{K}$-basis of the vector space $\mathbb{R}_L/(I \cap \mathbb{R}_L)$ is returned by `ResidueClassBasis` as a list if it is finite or is enumerated by a formal power series [10] in case it is infinite. For examples of how to apply these two commands, cf. Section 4.

Given an affine (i.e. inhomogeneous) linear system of difference equations, a call of `CompCond` after the application of `JanetBasis` returns a generating set of compatibility conditions for the affine part of the system, i.e. necessary conditions for the right hand sides of the inhomogeneous system for solvability.

Moreover, combinatorial devices to compute the Hilbert series and polynomial and function etc. [4] are included in LDA.

For the application of LDA to the reduction of Feynman integrals, a couple of special commands were implemented to impose further relations on the master integrals: By means of `AddRelation` an infinite sequence of master integrals parametrized by indeterminates which are not contained in the list of independent variables is set to zero. Subsequent calls of `InvReduce` and `ResidueClassBasis` take these additional relations into account (cf. Section 4).

Finally, LDA provides several tools for dealing with difference operators. Difference operators represented by polynomials can be applied to (lists of) expressions containing y^1, ..., y^m as functions of the independent variables. Conversely, the difference operators can be extracted from systems of difference equations. Leading terms of difference equations can be selected.

2.2. Implementation details

We consider difference rings containing shift operators which act in one direction only. If a linear system of difference equations is given containing functions shifted in both directions, then the system needs to be shifted by the maximal negative shift in order to obtain a difference system with shifts in one direction only. However, LDA allows to change the shift direction globally.

Unnecessary computations of involutive reductions to zero are avoided using the four involutive criteria described in [8], [9]. Fine-tuning is possible by selecting the criteria individually.

The implemented monomial orders/rankings are the (block) degree-reverse-lexicographical and the lexicographical one. In the case of more than one dependent variable, priority of comparison can be either given to the difference operators ("term over position") or to the dependent variables ("position over term"/elimination ranking).

The ranking is controlled via options given to each command separately. The other options described above can be set for the entire LDA session using the command `LDAOptions` which also allows to select Janet or Janet-like division.

3. GENERATION OF FINITE DIFFERENCE SCHEMES FOR PDES

We consider the Laplace equation $u_{xx} + u_{yy} = 0$ and rewrite it as the conservation law

$$\oint_\Gamma -u_y dx + u_x dy = 0.$$

Adding the integral relations

$$\int_{x_j}^{x_{j+2}} u_x dx = u(x_{j+2}, y) - u(x_j, y),$$

$$\int_{y_k}^{y_{k+2}} u_y dx = u(x, y_{k+2}) - u(x, y_k)$$

and using the midpoint integration method we obtain the following discrete system:

$$\begin{cases} -(\theta_x - \theta_x \theta_y^2) \circ u_y + (\theta_x^2 \theta_y - \theta_y) \circ u_x = 0, \\ 2\triangle h\, \theta_x \circ u_x - (\theta_x^2 - 1) \circ u = 0, \\ 2\triangle h\, \theta_y \circ u_y - (\theta_y^2 - 1) \circ u = 0, \end{cases} \quad (1)$$

where θ_x and θ_y are the right-shift operators w.r.t. x and y, e.g., $\theta_x \circ u_y(x, y) = u_y(x+1, y)$.

Table 1
Main commands of LDA

`JanetBasis`	Compute Janet(-like Gröbner) basis
`InvReduce`	Involutive / $\mathcal{J}-$reduction modulo Janet(-like Gröbner) basis
`CompCond`	Return compatibility conditions for inhomogeneous system
`HilbertSeries` etc.	Combinatorial devices
`Pol2Shift`, `Shift2Pol`	Conversion between shift operators and equations

Some interpretations of commands for the reduction of Feynman integrals:

`ResidueClassBasis`	Enumeration of the master integrals
`AddRelation`	Definition of additional relations for master integrals
`ResidueClassRelations`	Return the relations defined for master integrals

We show how to use LDA to find a finite difference scheme for the Laplace equation:

```
>  with(LDA):
```

We enter the independent and the dependent variables for the problem ($ux > uy > u$):

```
>  ivar:=[x,y]:  dvar:=[ux,uy,u]:
```

Next, we translate (1) into the input format of `JanetBasis`. Note that one can in general use `AppShiftOp` to apply a difference operator given as a polynomial similar to the ones in (1) to a difference polynomial.

```
>  L:=[2*h*ux(x+1,y)-u(x+2,y)+u(x,y),
>  2*h*uy(x,y+1)-u(x,y+2)+u(x,y),
>  2*h*(ux(x+2,y+1)-ux(x,y+1))+
>  2*h*(uy(x+1,y+2)-uy(x+1,y))]:
```

Then we compute the minimal Janet basis of the linear difference ideal generated by L w.r.t. a ranking which compares the dependent variables prior to the corresponding difference monomials ("position over term" order; this ranking is chosen when using the option 2 as below). The least element of this Janet basis is by construction a difference polynomial which does not contain any monomial in ux and uy because $ux > uy > u$.

```
>  JanetBasis(L,ivar,dvar,2)[1][1];
```

$u(x + 4,\, y + 2) - 4\,u(x + 2,\, y + 2) + u(x,\, y + 2)$
$+u(x + 2,\, y + 4) + u(x + 2,\, y)$

The computation takes less than one second of time on a Pentium III (1 GHz).

Dividing this difference polynomial by $4h^2$ we obtain the following finite difference scheme:

$$D_j^2(u_{jk}) + D_k^2(u_{jk}) = 0,$$

where

$$D_j^2(u_{jk}) = \frac{u_{j+2\,k} - 2u_{j\,k} + u_{j-2\,k}}{4h^2}$$

and

$$D_k^2(u_{jk}) = \frac{u_{j\,k+2} - 2u_{j\,k} + u_{j\,k-2}}{4h^2}$$

are discrete approximations of the second order partial derivatives occurring in Laplace's equation.

4. REDUCTION OF FEYNMAN INTEGRALS

In order to demonstrate how to use LDA for the reduction of Feynman integrals, we consider a simple one-loop propagator type scalar integral with one massive and another massless particle:

$$f(k,n) := I_{k,n} = \frac{1}{i\pi^{d/2}} \int \frac{d^d s}{P_{s-q,m}^k P_{s,0}^n}.$$

(Here k, n are the exponents of the propagators.)

The basis integrals for this example and the corresponding reduction formulae were found and studied by several authors (see, e.g., [11,12]). Here we apply the Gröbner basis method, as implemented in LDA, directly to the recurrence re-

lations which have the form:

$$\begin{cases} [d - 2k - n - 2m^2k\mathbf{1}^+ - \\ n\mathbf{2}^+(\mathbf{1}^- - q^2 + m^2)]\,f(k+1, n+1) = 0, \\ [n - k - k\mathbf{1}^+(q^2 + m^2 - \mathbf{2}^-) - \\ n\mathbf{2}^+(\mathbf{1}^- - q^2 + m^2)]\,f(k+1, n+1) = 0, \end{cases} \quad (2)$$

where

$$\mathbf{1}^{\pm} f(k, n) = f(k{\pm}1, n), \; \mathbf{2}^{\pm} f(k, n) = f(k, n{\pm}1).$$

In addition, it is known that

$$f(k + i, n + j) = 0 \quad \forall\, i \le 0 \quad \forall\, j \qquad (3)$$

which we will take into account later.

```
>  ivar:=[k,n]:  dvar:=[f]:
```

We enter the recurrence relations (2):

```
>  L:=[(d-2*k-n)*f(k+1,n+1)-2*m^2*k*
>  f(k+2,n+1)-n*f(k,n+2)-n*(m^2-q^2)*
>  f(k+1,n+2),   (n-k)*f(k+1,n+1)-
>  k*(q^2+m^2)*f(k+2,n+1)+k*f(k+2,n)-
>  n*f(k,n+2)-n*(m^2-q^2)*f(k+1,n+2)]:
>  JanetBasis(L,ivar,dvar):
```

Again, the computation time is less than one second. Now, the master integrals are given by:

```
>  ResidueClassBasis(ivar,dvar);
```

$$[f(k, n), f(k, n+1), f(k+1, n),$$
$$f(k, n+2), f(k+1, n+1), f(k+2, n)]$$

(3) implies additional relations on the master integrals. (Here, j is recognized as not being contained in `ivar` and thus serves as a parameter to define the additional relations.)

```
>  AddRelation(f(k,n+j)=0,ivar,dvar):
```

The list of master integrals now becomes:

```
>  ResidueClassBasis(ivar,dvar);
```
$$[f(k+1, n), f(k+1, n+1), f(k+2, n)]$$

Next, we recompute the Janet basis for $m = 0$:

```
>  m:=0:  J:=JanetBasis(L,ivar,dvar):
```

For the special case where $m = 0$, we impose the relation $f(k + i, n) = 0$ for all i:

```
>  AddRelation(f(k+i,n)=0,ivar,dvar):
```

Now, we are left with one master integral:

```
>  ResidueClassBasis(ivar,dvar);
```
$$[f(k+1, n+1)]$$

We reduce $f(k + 2, n + 3)$ modulo J taking also the additionally imposed relations on the master integrals into account. (Here, the option "F" lets `InvReduce` return the result in factorized form.)

```
>  InvReduce(f(k+2,n+3),J,"F");
```

$$-((2n + 4 - d + 2k)(2n + 2 - d + 2k)$$
$$(2k + n - d)(n + 3 - d + k)$$
$$(n + 2 - d + k)\,\mathrm{f}(k+1, n+1))/((n+1)$$
$$(2n - d + 4)n\,q^6\,k\,(d - 2k - 2))$$

Using `ResidueClassRelations` one can display the relations imposed on the master integrals:

```
>  ResidueClassRelations(ivar,dvar,[i,j]);
```
$$[\mathrm{f}(k, n + j), \mathrm{f}(k + i, n)]$$

The difference operators occurring in the last result can be extracted as polynomials in δ_k, δ_n:

```
>  Shift2Pol(%,ivar,dvar,
>  [delta[k],delta[n]]);
```
$$[\delta_n{}^j, \delta_k{}^i]$$

5. CONCLUSION

We presented the Maple package LDA implementing the Janet/Janet-like division algorithm for computing Gröbner bases of linear difference ideals. Generation of difference schemes for linear PDEs and reduction of loop Feynman integrals are important applications of the package.

These two kinds of applications were illustrated by rather simple examples. The first difference system (discrete Laplace equation and integral relations) contains two independent variables (x, y) and three dependent variables (u, u_x, u_y). The second system (recurrence relations for one-loop Feynman integral) also contains two independent variables/indices (k, n), but the only dependent variable f. The second system, however, is computationally slightly harder than the first one because of explicit dependence of the recurrence

relations on the indices and three parameters (d, m^2, q^2) involved in the dependence on indices.

Dependence on index variables and parameters is an attribute of recurrence relations for Feynman integrals. Similar dependence may occur in the generation of difference schemes for PDEs with variable coefficients containing parameters. Theoretically established exponential and superexponential (depends on the ideal and ordering) complexity of constructing polynomial Gröbner bases implies that construction of difference Gröbner bases is at least exponentially hard in the number of independent variables (indices). Besides, in the presence of parameters the volume of computation grows very rapidly as the number of parameters increases.

Thus, for successful application of the Gröbner basis technique to multiloop Feynman integrals with masses and to multidimensional PDEs with multiparametric variable coefficients we are going not only to improve our Maple code but also to implement the algorithms for computing Janet and/or Janet-like difference bases in C++ as a special module of the open source software available on the Web page `http://invo.jinr.ru`.

6. ACKNOWLEDGEMENTS

REFERENCES

1. V.P.Gerdt. *Gröbner Bases in Perturbative Calculations.* Nuclear Physics B (Proc. Suppl.) 135, 2004, pp. 232–237. arXiv:/hep-ph/0501053.

2. V.V.Mozzhilkin, Yu.A.Blinkov. *Methods of Constructing Difference Schemes in Gas Dynamics.* Transactions of Saratov University, 1(2), 2001, pp. 145–156 (in Russian).

3. V.P.Gerdt, Yu.A.Blinkov, V.V.Mozzhilkin. *Linear Difference Ideals and Generation of Difference Schemes for PDEs.* Symmetry, Integrability and Geometry: Methods and Applications (SIGMA), Institute of Mathematics, Kiev, 2005, to appear.

4. Yu. A. Blinkov, V. P. Gerdt, C. F. Cid, W.Plesken, D.Robertz. *The Maple Package "Janet": I. Polynomial Systems.* Computer Algebra in Scientific Computing / CASC 2003, V. G. Ganzha, E. W. Mayr, and E. V. Vorozhtsov (Eds.), Institute of Informatics, Technical University of Munich, Garching, 2003, pp. 31–40; *II. Linear Partial Differential Equations*, ibid., pp. 41–54.

5. V.P.Gerdt. *Involutive Algorithms for Computing Gröbner Bases.* Computational commutative and non-commutative algebraic geometry, IOS Press, Amsterdam, 2005, pp. 199–225. arXiv:math.AC/0501111, 2005.

6. V.P.Gerdt, Yu.A.Blinkov. *Janet-like Monomial Division.* Computer Algebra in Scientific Computing / CASC 2005, V. G. Ganzha, E. W. Mayr, E. V. Vorozhtsov (Eds.), Springer-Verlag, Berlin, 2005, pp. 174–183; *Janet-like Gröbner Bases*, ibid., pp. 184-195.

7. V.P.Gerdt. *On Computation of Gröbner Bases for Linear Difference Systems.* This volume.

8. V.P.Gerdt, D.A.Yanovich. *Experimental Analysis of Involutive Criteria.* Algorithmic Algebra and Logic 2005, A.Dolzmann, A.Seidl, T.Sturm (Eds.), BOD Norderstedt, Germany, 2005, pp. 105–109.

9. J.Apel, R.Hemmecke. *Detecting unnecessary reductions in an involutive basis computation*, RISC Linz Report Series 02-22, 2002.

10. W.Plesken, D.Robertz. *Janet's approach to presentations and resolutions for polynomials and linear pdes.* Archiv der Mathematik, 84 (1), 2005, pp. 22–37.

11. O.V.Tarasov. *Reduction of Feynman graph amplitudes to a minimal set of basic integrals.* Acta Physica Polonica B29, 1998, pp. 2655–2666. arXiv:/hep-ph/9812250

12. V.A.Smirnov. *Evaluating Feynman Integrals*, STMP 211, Springer, Berlin, 2004.