EPJ Web of Conferences **173**, 05018 (2018) https://doi.org/10.1051/epjconf/201817305018
Mathematical Modeling and Computational Physics 2017

Computing Gröbner and Involutive Bases for Linear Systems of Difference Equations

Denis Yanovich[1,⋆]

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

Abstract. The computation of involutive bases and Gröbner bases for linear systems of difference equations is solved and its importance for physical and mathematical problems is discussed. The algorithm and issues concerning its implementation in C are presented and calculation times are compared with the competing programs. The paper ends with consideration on the parallel version of this implementation and its scalability.

1 Introduction

Recently there was considerable interest in applying difference Gröbner bases to the reduction of multiloop Feynman integrals to minimal sets of basis integrals [1]. The related systems of linear difference polynomial equations are difficult to transform into the Gröbner basis form with available software. In this note we present a competitive software implementing the involutive approach to compute the basis form.

1.1 Elements of Difference Algebra [2]

Let $\{y^1,\dots,y^m\}$ be the set of indeterminates – m functions (dependent variables) of n variables $x_1,\dots,x_n$ (independent variables), and $\{\theta_1,\dots,\theta_n\}$ be the set of mutually commuting difference operators:

$$(\theta_i \circ y^j)(x_1,\dots,x_n) = y^j(x_1,\dots,x_i+1,\dots,x_n).$$

A difference ring R with differences $\theta_1,\dots,\theta_n$ is a commutative ring:

$$\theta_i\theta_j = \theta_j\theta_i, \qquad \theta_i \circ (f+g) = \theta_t \circ f + \theta_i \circ g, \qquad \theta_i \circ (fg) = (\theta_i \circ f)(\theta_i \circ g).$$

Let $\mathcal{R}$ be the difference ring of polynomials over field $\mathcal{K}$ in variables

$$\{\theta^\mu \circ y^k \mid \mu \in \mathbb{Z}^n_{\geq 0},\ k = 1,\dots,m\}.$$

Let $\mathcal{R}_\mathcal{L}$ be the set of linear polynomials from $\mathcal{R}$ and let

$$\Theta = \{\theta^\mu \mid \mu \in \mathbb{Z}^n_{\geq 0}\}, \qquad \deg_i(\theta^\mu \circ y^k) = \mu_i, \qquad \deg(\theta^\mu \circ y^k) = \sum_{i=1}^{n} \mu_i.$$

⋆e-mail: yan@jinr.ru

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (http://creativecommons.org/licenses/by/4.0/).

A difference ideal is an ideal $\mathcal{J} \subseteq \mathcal{R}$ closed under the action of any operator from Θ. For $F \in \mathcal{R}$, the smallest difference ideal containing F will be denoted by $\mathrm{Id}(F)$. If for an ideal $\mathcal{J}$ there is $F \in \mathcal{R}_{\mathcal{L}}$ such that $\mathcal{J} = \mathrm{Id}(F)$, then $\mathcal{J}$ is a linear difference ideal.

A total ordering $\succ$ on the set of $\theta^\mu \circ y^j$ is a ranking if for all i, j, k, μ, ν the following holds:

$$\theta_i\theta^\mu \circ y^j \succ \theta^\mu \circ y^j,$$
$$\theta^\mu \circ y^j \succ \theta^\nu \circ y^k \Longleftrightarrow \theta_i\theta^\mu \circ y^j \succ \theta_i\theta^\nu \circ y^k.$$

If $|\mu| > |\nu|$ implies $\theta^\mu \circ y^j \succ \theta^\nu \circ y^k$ for all j, k, then the ranking is orderly. If $j > k$ implies $\theta^\mu \circ y^j \succ \theta^\nu \circ y^k$ for all μ, ν, then the ranking is elimination.

Given a ranking $\succ$, a linear polynomial $f \in \mathcal{R}_{\mathcal{L}} \setminus \{0\}$ has the leading term $a\vartheta \circ y^j$, $\vartheta \in \Theta, a \in \mathcal{K}$, where $\vartheta \circ y^j$ is maximal w.r.t. $\succ$ among all $\theta^\mu \circ y^k$ which appear with nonzero coefficient in f. $\mathrm{lc}(f) = a \in \mathcal{K} \setminus \{0\}$ is the leading coefficient and $\mathrm{lm}(f) = \vartheta \circ y^j$ is the leading monomial.

A ranking acts in $\mathcal{R}_{\mathcal{L}}$ as a monomial order. If $F \subseteq \mathcal{R}_{\mathcal{L}} \setminus \{0\}$, $\mathrm{lm}(F)$ will denote the set of the leading monomials and $\mathrm{lm}_j(F)$ will denote its subset for the indeterminate y^j. Thus,

$$\mathrm{lm}(F) = \bigcup_{j=1}^{m} \mathrm{lm}_j(F).$$

1.2 Gröbner and Involutive Bases of a Difference Ideal [2–4]

Given a nonzero linear difference ideal $\mathcal{J} = \mathrm{Id}(\mathcal{G})$ and a ranking $\succ$, the ideal generating the set $G = \{g_1, \ldots, g_s\} \subset \mathcal{R}_{\mathcal{L}}$ is a *Gröbner basis* of $\mathcal{J}$ with respect to $\succ$ if for all $f \in \mathcal{J} \cap \mathcal{R}_{\mathcal{L}} \setminus \{0\}$:

$$\exists g \in G,\ \theta \in \Theta : \mathrm{lm}(f) = \theta \circ \mathrm{lm}(g).$$

It follows that $f \in \mathcal{J} \setminus \{0\}$ *is reducible modulo* G:

$$f \underset{g}{\to} f' := f - \mathrm{lc}(f)\theta \circ (g/\,\mathrm{lc}(g)), \quad f' \in \mathcal{J}.$$

If $f' \neq 0$, then it is again reducible modulo G, and, by repeating the reduction, in finitely many steps we obtain

$$f \underset{G}{\to} 0.$$

Similarly, a nonzero polynomial $h \in \mathcal{R}_{\mathcal{L}}$, the terms of which are reducible (if any) modulo a set $F \subset \mathcal{R}_{\mathcal{L}}$, can be reduced to an irreducible polynomial $\bar{h}$, which is said to be *in normal form modulo* F (denotation: $\bar{h} = NF(h, F)$).

Replacing the conventional division with the involutive one [3] are obtained definitions of the *involutive basis* and *involutive normal form* which have several major advantages as compared to the Gröbner basis computation: automatic avoidance of some unnecessary reductions, smoother arithmetics growth during calculation and natural parallelism.

The author used the Janet division as involutive, but there exist theoretically better divisions, for example Janet-like [2].

2 Algorithm

In the Algorithm 1 below the computations at the lines 3–8, which consume about 90% of programm runtime are mutually independent, so they may be started simultaneously. This is a source of natural parallelism of the involutive algorithm [5–7].

Algorithm 1 Sequental Computation of Involutive Basis

Input: F, a set of polynomials from $\mathcal{R}_{\mathcal{L}}$, L, an involutive division, $\succ$, a monomial ordering
Output: T, an involutive basis of F with respect to $\succ$

```
1:  T := ∅;  Q := F
2:  while Q ≠ ∅ do
3:     for all { f ∈ Q | lm(f) = min_≻(lm(Q)) } do
4:        Q := Q \ {f};   f' := NF_L(f, T)
5:        if f' ≠ 0 then
6:           Q := Q ∪ {f'}
7:        fi
8:     od
9:     T := T ∪ { p | lm(p) = min_≻(lm(Q)) }
10:    Q := Q \ {p};  Q := Q ∪ { θ^μ ∘ p, μ ∈ NM_L(p, T) }
11:    if lm(p) was reduced on step 4 then
12:       for all { r ∈ T | lm(r) = θ^ν ∘ lm(p) } do
13:          Q := Q ∪ {r};   T := T \ {r}
14:       od
15:    fi
16: od
```

3 Implementation Issues

This algorithm was implemented as a program written in C language. The initial system of linear difference equations rewrites the input data into the form of a polynomial system with parametric coefficients (independent variables) and power products of the forward shift operators $\theta^{\mu}, |\mu| \geq 0$ applied to the dependent variable (sec. 1.1). For computations of the greatest common divisor for reduced polynomials with parametric coefficients our implementation uses the well-known multi-modular Brown algorithm [8], arbitrary precision arithmetics uses *libgmp*.

4 Benchmarking

Due to their high practical value for the computation of Feynman integrals [1] we shall take some systems of difference equations as benchmarks and examples. As a competitive software, the Maple package LDA [2] was chosen. The comparative timings of sequental programs are shown in Table 1. The most difficult computed example, *Smirnov3*, was used as a benchmark for the parallel version of our program (Table 2).

Smirnov1 [1]: Integral relations for the one-loop massless propagator

$$\begin{aligned}&(d-2a_1-a_2)f(a_1,a_2)-a_2f(a_1-1,a_2)+a_2q^2f(a_1,a_2+1),\\&(a_2-a_1)f(a_1+1,a_2+1)-a_1q^2f(a_1+1,a_2)+a_1f(a_1+1,a_2-1)\\&-a_2f(a_1-1,a_2+1)+a_2q^2f(a_1,a_2+1).\end{aligned}$$

Smirnov2 [1]: Integral relations for the one-loop propagator with mass

$$\begin{aligned}&(d-2a_1-a_2)f(a_1,a_2)-2m^2a_1f(a_1+1,a_2)-m^2a_2f(a_1,a_2+1)\\&+q^2a_2f(a_1,a_2+1)-a_2f(a_1-1,a_2+1),\\&(a_2-a_1)f(a_1,a_2)-m^2a_1f(a_1+1,a_2)-q^2a1f(a_1+1,a_2)\\&-m^2a_2f(a_1,a_2+1)+q^2a_2f(a_1,a_2+1)-a_2f(a_1-1,a_2+1)+a_1f(a_1+1,a_2-1).\end{aligned}$$

Smirnov3 [1] consists of six integral relations for two-loop massless propagators and utilizes five independent variables and two parameters.

Table 1. Sequental programs, time in seconds@Intel Core i5-3470

Benchmark	JB (present)	LDA
Smirnov1	16559.10	∞
Smirnov2	0.02	0.22
Smirnov3	3629.42	∞

Table 2. Parallel program, *Smirnov3* example, time in seconds@Hybrilit cluster [9], Intel Xeon E5-2695 v2

threads	compute time	speedup	CPU-utilization
1	2585.3		
3	1460.65	1.77	0.59
8	1555.57	1.66	0.21
16	507.05	5.10	0.32

5 Conclusion

Using a similar algorithmic background, our implementation demonstrates significantly better timings in comparison with the Maple implementation. It has thus the potential for computing relatively large problems. The parallel version shows rather good CPU-utilization, but it is not the final result. Hope to achieve better results in the future.

Acknowledgements

The work is supported in part by the Russian Foundation for Basic Research (grant 16-01-00080). The author thanks Vladimir Gerdt for the problem statement and useful remarks.

References

[1] A.V. Smirnov and V.A. Smirnov, Journal of High Energy Physics **0601**, 001 (2006)
[2] V.P. Gerdt, and D. Robertz, Computer Science Journal of Moldova **20**, no. 2 (59), 203–226 (2012)
[3] V.P. Gerdt and Yu.A. Blinkov, Mathematics and Computers in Simulation **45**, 519–542 (1998)
[4] V.P. Gerdt, Physics of Particles and Nuclei Letters **5**, no. 3, 248–254 (2008)
[5] D.A. Yanovich, Programming and Computer Software **28**, no. 2, 66–69 (2002)
[6] V.P. Gerdt and D.A. Yanovich, Programming and Computer Software **31**, no. 2, 73–80 (2005)
[7] D.A. Yanovich, Vestnik Rossijskogo universiteta druzhby narodov. Seriya Matematika, informatika, fizika **3.2**, 19–24 (2010)
[8] W.S. Brown, Journal of the ACM **18**, 476–504 (1971)
[9] `http://hybrilit.jinr.ru`