Summing-up Involutive Bases Computations Experience

Denis A. Yanovich

Joint Institute for Nuclear Research, Dubna, Russia

PCA2023, 17 April 2023

In Memory of Professor V.P.Gerdt

21.01.1947 – 05.01.2021

Involutive Division

- Zharkov, A. Yu., Blinkov, Yu. A.: Involutive approach to investigating polynomial systems. Math. Comp. Simul., 42 (1996), 323-332
- Gerdt, V. P., Blinkov, Yu. A.: Involutive Bases of Polynomial Ideals. Math. Comp. Simul. 45 (1998) 519–542
- Gerdt, V. P., Blinkov, Yu. A.: Minimal Involutive Bases. Math. Comp. Simul. 45 (1998) 543–560

Let's somehow choose some variables $M(u, U)$ of monomial u from monomial set U and call this subsets *multiplicative variables*.

Let's narrow conventional division: allow division only by variables from $M(u, U)$. This would be *involutive division* L.

Involutive Division Traits

- global/local
- noetherian
- continuous
- constructive

Janet Division: Example

$$U = \{x^2y, xz, y^2, yz, z^2\}, (x \succ y \succ z)$$

Monomial	Nonmultiplicative	Multiplicative
x^2y	–	x, y, z
xz	x	y, z
y^2	x	y, z
yz	x, y	z
z^2	x, y	z

Janet Division

Given a monomial order $\succ$, a finite monomial set U, and a monomial $u \in U$, the *Janet separation* of variables into $M_J(u, U)$ and $NM_J(u, U)$ is defined as follows:
For each $1 \leq i \leq n$ divide U into groups labeled by non-negative integers $d_1, \ldots, d_i$

$$[d_1, \ldots, d_i] = \{v \ \in F \mid d_j = \deg_j(v),\ 1 \leq j \leq i\}.$$

x_1 is (Janet) multiplicative for $u \in U$ if

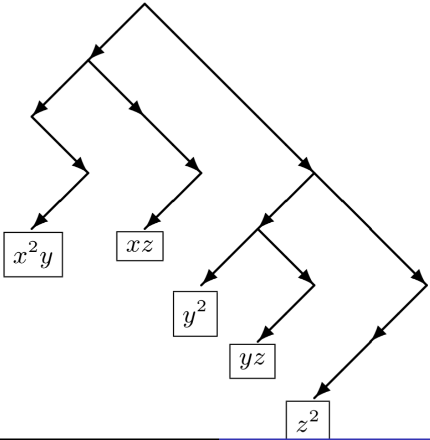
$$\deg_1(u) = \max\{\deg_1(v) \mid v \in U\}$$

For $i > 1$ x_i is multiplicative for $u \in [d_1, \ldots, d_{i-1}]$ when

$$\deg_i(u) = \max\{\deg_i(v) \mid v \in [d_1, \ldots, d_{i-1}]\}$$

Janet Tree

Gerdt, V.P., Blinkov, Y., Yanovich, D.: Construction of Janet bases I: Monomial bases. In: Ghanza, V., Mayr, E., Vorozhtsov, E. (eds.) Computer Algebra in Scientific Computing, CASC 2001, pp. 233–247. Springer-Verlag, Berlin (2001)



Involutive Ideal

Definition, monomial

$$(\forall u \in U)\ (\forall x \in NM_L(u,U))\ (\exists v \in U : v|_L(u \cdot x))$$

Involutive normal form

$$NF_L(p,F) = p - \sum_{ij} \alpha_{ij} m_{ij} g_j$$

$$\alpha_{ij} \in \mathbb{K},\ \ g_j \in F,\ \ m_{ij} \in M(\mathrm{lm}(g_j), \mathrm{lm}(F)),\ \ \mathrm{lm}(m_{ij} g_j) \preceq \mathrm{lm}(p).$$

Involutive Polynomial Ideal

Given an ideal $I \subset \mathbb{R}$, an involutive division L and monomial order $\succ$, a finite $L-$autoreduced subset $T \subset \mathbb{R}$ generating I is called its *$L-$(involutive) basis* if

$$(\forall g \in I)\ (\exists f \in T)\ [\ \mathrm{lm}(f) \mid_L \mathrm{lm}(g)\ ]$$

If division L is continuous this is equivalent to

$$(\ \forall f \in T\ )\ (\ \forall x_i \in NM_L(\mathrm{lm}(f), \mathrm{lm}(T))\ )\ \ [\ NF_L(x_i \cdot f, T) = 0\ ]$$

The product $x_i \cdot f$ of polynomial $f \in T$ and $x_i \in NM_L(f, T)$ is called *nonmultiplicative prolongation* of f, and construction of involutive bases is often called *completion*.

Involutive and Gröbner bases are examples of *Canonical Form*

Involutive Basis Computation Algorithm

Input: F, L, $\prec$
Output: T – involutive basis of F

$T := \emptyset \;\; Q := F$
while $Q \neq \emptyset$ **do**
$\quad T := T \cup \{p \mid lm(p) = min(lm(Q)),\;\; NF_L(p, T) \neq 0\} \Leftarrow$!!!
$\quad Q := Q \setminus \{p\}$
$\quad Q := Q \cup \{p' \mid \forall x \notin M(lm(p), lm(T)) : p' = p \cdot x\}$
$\quad$ **for all** $\{\; r \in T \mid lm(r) \sqsupset \mathrm{lm}(p) \;\}$ **do**
$\quad\quad Q := Q \cup \{r\}; \qquad T := T \setminus \{r\}$
$\quad$ **od**
od

Parallel Algorithms

- Yanovich, D. A.: Parallelization of an Algorithm for Computation of Involutive Janet Bases. Prog. and Comp. Soft., 28(2), 2002, pp. 66-69
- Gerdt V.P., Yanovich D.A.: Parallelism in Computing Janet Bases // Proceedings of the Workshop on Under- and Overdetermined Systems of Algebraic or Differential Equations (Karlsruhe, March 18-19, 2002), J.Calmet, M.Hausdorf, W.M.Seiler (Eds.). Institute of Algorithms and Cognitive Systems, University of Karlsruhe. 2002, P.47-56.

Parallel Algorithm: SMP

$T := \emptyset \;\; Q := F$, F – initial polynomial set, T – basis
while $Q \neq \emptyset$ **do**
  $S := \emptyset \;\; P := \{\, q_i \in Q \mid i \leq K_{thr}, q_i - min \in Q \,\}$
  $Q := Q \setminus P$
  $S := NF_{Lead}(P)$ using K_{thr} threads
  $Q := Q \cup S$
  $T := T \cup \{p \mid lm(pol(p)) = min(lm(Q))\} \;\; Q := Q \setminus \{p\}$
  $Q := Q \cup \{p \cdot x_i \mid x_i \in \mathrm{nmp}(p)\}$
  if $\mathrm{lm}(\mathrm{pol}(p)) = \mathrm{anc}(p)$ **then**
    for all $\{\, r \in T \mid lm(\mathrm{pol}(r)) \sqsupset \mathrm{lm}(\mathrm{pol}(p)) \,\}$ **do**
      $Q := Q \cup \{r\}; \qquad T := T \setminus \{r\}$
    od
    $S := NF_{Full}(T)$ using K_{thr} threads
    $T := S$
  fi
od

Parallel Algorithm: Distributed, Main

Yanovich, D.A.: Reduction-Level Parallel Computations of Gröbner and Janet Bases. *Bulletin of Peoples' Friendship University of Russia*, Mathematics. Information Sciences. Physics. No.3, Issue 2 (2010), pp. 19–24.

$GroupSize$:= number of computational nodes
$MyRank$:= rank in group
$T := \emptyset \quad Q := F$
if $MyRank = 0$ **then**
    distribute Q to Q_i
fi
while $Q_i \neq \emptyset \mid i = 0..GroupSize$ **do**
    Algorithm Step
od

Parallel Algorithm: Distributed, Step 1/2

$S := \emptyset \;\; P := \{ \; q_i \in Q \; | q_i - min \in Q \; \}$
$Q := Q \setminus P \quad S := NF_{Lead}(P) \quad Q := Q \cup S$
$h := \{p \in Q \mid lm(pol(p)) = min(lm(Q))\} \;\; Q := Q \setminus \{p\}$
gather $h_i \;\; i = 0..GroupSize$
choose $h_j \mid lm(pol(h_j)) = min(lm(\{h_i\}))$
if $MyRank = j$ **then**
    broadcast h_j **as** h – new basis element
else
    $Q := Q \cup h_{MyRank}$
fi
$T := T \cup h$
if $MyRank = 0$ **then**
    $S := \{h \cdot x_i \mid x_i \in \mathrm{nmp}(h)\}$
    distribute S by Q_i
fi

Parallel Algorithm: Distributed, Step 2/2

if $\mathrm{lm}(\mathrm{pol}(h)) = \mathrm{anc}(h)$ **then**
  for all $\{\ r \in T \mid lm(\mathrm{pol}(r)) \sqsupset \mathrm{lm}(\mathrm{pol}(h))\ \}$ **do**
    $T := T \setminus \{r\}$
    if $MyRank = 0$ **then**
      $Q := Q \cup \{r\}$
    fi
  od
  $S := NF_{Full}(T)$
  $T := S$
fi

Multimodal Basis Computation: Problem

Yanovich, D.A.: Parallel modular computation of Gröbner and involutive bases. Program Comput Soft 39 (2013), 110–113

Motivation

1. Method to avoid intermediate coefficient growth
2. Natural and effective parallelism

Difficulties

1. Unlucky primes
2. One cannot directly lift the polynomial with $\mathbb{Z}$-coefficients from its modular images: each of them is multiplied by the unknown common modular factor $c_p \cdot f_p$, different for every p

Multimodular Basis Computation: Theory

Chinese Reminder Theorem

Having modular images $c_1, \cdots, c_n$ of a number c with respective modules $m_1, \cdots, m_n$ one can construct $c/\mathbb{Z}_M, \;\; M = \prod m_i$

Farey Fractions

The Farey fractions $\mathbb{F}_N$ is the set of all fractions in lowest terms between 0 and 1 whose denominators do not exceed N, arranged in order of magnitude. There is a one-to-one mapping between $\mathbb{F}_N$ and $0, 1, \cdots, p-1$, where $N \leq \sqrt{(p-1)/2}$ (P. Kornerup, R. T. Gregory: Mapping Integers and Hensel Codes Onto Farey Fractions)

Example

$F_5 = \{\frac{0}{1}, \frac{1}{5}, \frac{1}{4}, \frac{1}{3}, \frac{2}{5}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}, \frac{3}{4}, \frac{4}{5}, \frac{1}{1}\}$

Multimodular Basis Computation: Algorithm Outline

- Get the polynomial system in $\mathbb{Z}$ and compute several bases (Gröbner or involutive) in $\mathbb{Z}_p$ over $m_1, \cdots, m_n$
- Select images modulo lucky primes
- Reconstruct them to $\mathbb{Z}_M, \; M = \prod m_i$
- Map all modular coefficients to $\mathbb{F}_M$ Farey fractions and integer numerator. If M is sufficiently large, we will have all rational coefficients from the basis of the original system reconstructed.

System of Linear Difference Equations

- Gerdt, V.P.: Gröbner Bases Applied to Systems of Linear Difference Equations, Physics of Particles and Nuclei Letters, **5**, no. 3 (2008), pp. 248-254
- Gerdt, V.P., Robertz, D.: Computation of Difference Gröbner Bases, Computer Science Journal of Moldova, **20**, no. 2(59) (2012), pp. 203-226
- Yanovich, D.A.: Computing Gröbner and Involutive Bases for Linear Systems of Difference Equations, EPJ Web Conf. Volume 173, 2018

Difference Ideals

Difference operators

Let indeterminates $y^1, \ldots, y^m$ be functions of variables $x_1, \ldots, x_n$
Let $\theta_1, \ldots, \theta_n$ be differences

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

Difference ring properties

$$\theta_i \theta_j = \theta_j \theta_i$$
$$\theta_i \circ (f + g) = \theta_i \circ f + \theta_i \circ g$$
$$\theta_i \circ (fg) = (\theta_i \circ f)(\theta_i \circ g)$$

Difference Ideals

Monomial ordering – ranking

$\theta_i \theta^\mu \circ y^j \succ \theta^\mu \circ y^j$
$\theta^\mu \circ y^j \succ \theta^\nu \circ y^k \Leftrightarrow \theta_i \theta^\mu \circ y^j \succ \theta_i \theta^\nu \circ y^k$

Having ordering we can define leading term, normal form of difference equation, Gröbner and involutive basis for systems much alike algebraic polynomial case

From algebraic to difference

Prolongation by variable $\Rightarrow$ multiplying by shift operator

Difference Ideals: Algorithm Outline

Input: $F,\ L,\ \prec$
Output: T – involutive basis of F

$T := \emptyset \quad Q := F$
while $Q \neq \emptyset$ **do**
    $T := T \cup \{p \mid lm(p) = min(lm(Q)),\ NF(p,T) \neq 0\}$
    $Q := Q \setminus \{p\},\ Q := Q \cup \theta^{\mu} \circ p, \mu \in NM(p,T)$
    if $\mathrm{lm}(p) == \mathrm{anc}(p)$ **then**
        for all $\{\ r \in T \mid lm(r) = \theta^{\mu} \circ \mathrm{lm}(p)\ \}$ **do**
            $Q := Q \cup \{r\}; \qquad T := T \setminus \{r\}$
        od
    fi
od

Tableau Data Structure

Yanovich, D.A.: Computation of Involutive and Gröbner Bases Using the Tableau Representation of Polynomials, Prog. and Comp. Soft., Volume 46 (2), 2020, pp.162-166

Monomials

Let's construct the all-monomials-index: one number corresponds to the one monomial. Let it be the number of the column in the big tableau

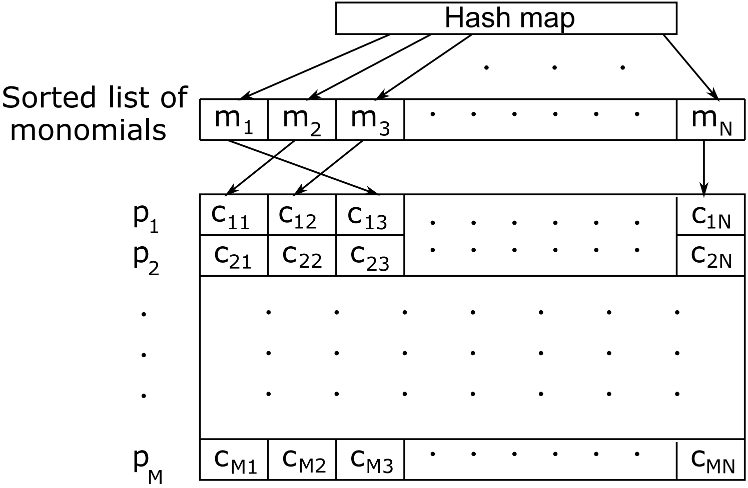
Polynomials

The one row of the big DENSE tableau corresponds to the one polynomial

Coefficients

Crossing of the column (monomial) and the row (polynomial) gives us term coefficient

Tableau polynomial set represenation



Parallel Computations: Tableau

Involutive Bases

When computing involutive bases one already have some sort of the natural parallelism (many reductions can be done in parallel)

Tableau

We don't have any strong coupling in the tableau: one can COMPUTE and STORE any part of the tableau separately

GPU Computations?

Yes! We don't have any pointer in our tableau, it's a perfect match for CUDA-like kernels.

What Next?

It was interesting and pleasant 25 years journey but all has it's ending, seems I have done all that I could:

- all repositories will be cleaned up and presented as public domain eventually


SO LONG AND...
Thanks for all the fish!