RESEARCH INSTITUTE FOR SYMBOLIC COMPUTATION | RISC

JOHANNES KEPLER UNIVERSITÄT LINZ

Report on Bachelor Thesis

Validating the Formalization of Theories and Algorithms of Polynomials in Computer Algebra by the Computer-Supported Checking of Finite Models

Julia Hofer

June 8, 2017

Seminar for Computer Algebra

The core of this Thesis

- Consider theories of polynomial algorithms,
- formalize those theories and
- enable a verification with RISCAL.

Formalization

- avoid possible errors without manually testing
- detailed mathematical models of algorithms
- checking of the underlying model of a system

RISCAL - RISC Algorithm Language

- formalization of mathematical theories and algorithms
- specification language, based on typed logic
  - Types
  - Predicates
  - Functions: implicit and explicit
  - Theorems
  - Procedures
- evaluation over finite domains
  - decidable statements
  - supports verification

Polynomial Algorithms

- addition
- subtraction
- multiplication
- division
- greatest common divisor
- resultants
- squarefree factorization

Polynomial Algorithms

univariate polynomials $p(x) = \sum_{i=0}^{n} p_i x^i$ and $q(x) = \sum_{i=0}^{m} q_i x^i$

- addition

$$p(x) + q(x) = \sum_{i=0}^{max(n,m)} (p_i + q_i) x^i$$

- subtraction

$$p(x) - q(x) = \sum_{i=0}^{max(n,m)} (p_i - q_i) x^i$$

- multiplication

$$p(x) \cdot q(x) = \sum_{l=0}^{n+m} (\sum_{i+j=l} p_i \cdot q_j) x^l$$

Polynomial Algorithms

- division

Algorithm POL_DIVK(in: a, b; out: q, r);
[$a, b \in K[x]$, $b \neq 0$; $q = \text{quot}(a, b)$, $r = \text{rem}(a, b)$. a and b are assumed to be in dense representation, the results q and r are likewise in dense representation]
1. $q :=$ [ ]; $a' := a$; $c := \text{lc}(b)$; $m := \deg(a')$; $n := \deg(b)$;
2. while $m \geq n$ do
   {$d := \text{lc}(a')/c$; $q := \text{CONS}(d, q)$; $a' := a' - d \cdot x^{m-n} \cdot b$;
   for $i = 1$ to $\min\{m - \deg(a') - 1, m - n\}$ do $q := \text{CONS}(0, q)$;
   $m := \deg(a')$};
3. $q := \text{INV}(q)$; $r := a'$; return.

Polynomial Algorithms

- greatest common divisor

Algorithm GCD_PRS(in: a, b; out: g);
$[a, b \in I[x]^*, g = \gcd(a, b)]$
1. if $\deg(a) \geq \deg(b)$
   then $\{f_1 := \mathrm{pp}(a);\ f_2 := \mathrm{pp}(b)\}$
   else $\{f_1 := \mathrm{pp}(b);\ f_2 := \mathrm{pp}(a)\}$;
2. $d := \gcd(\mathrm{cont}(a), \mathrm{cont}(b))$;
3. compute $f_3, \ldots, f_k, f_{k+1} = 0$ such that $f_1, f_2, \ldots, f_k, 0$ is a prs;
4. $g := d \cdot \mathrm{pp}(f_k)$; return.

Results of this Thesis

- formalization of polynomial algorithms with specification of the fundamental theories in RISCAL - types and conditions
- reasonable validation of the algorithms by model checking

Further work

- algorithms for bivariate and multivariate polynomials
  - bivariate: field of univariate polynomials
  - multivariate: recursive