Polynomial Data-Types

Marc Moreno Maza

CS 9652, September 13, 2017
Plan

Polynomial system solvers in action

Algebraic structures

Polynomials in algebra

Univariate polynomial data-type

Multivariate polynomial data-type

Polynomial representations by values

Straighth-line programs

Big integers
Plan

Polynomial system solvers in action

Algebraic structures

Polynomials in algebra

Univariate polynomial data-type

Multivariate polynomial data-type

Polynomial representations by values

Straight-line programs

Big integers
Real root isolation for zero-dimensional systems

\[(R \coloneqq \text{PolynomialRing}([x, y, z]); \quad F \coloneqq \{x^2 + y + z - 1, x + y^2 + z - 1, x + y + z^2 - 1\})\]

\[\text{polynomial_ring} \quad \{x^2 + y + z - 1, y^2 + x + z - 1, x + y + z^2 - 1\}\]

\[\text{dec} \coloneqq \text{Triangularize}(F, R); \quad \text{map}(\text{Display}, \text{dec}, R);\]

\[\text{regular_chain, regular_chain, regular_chain, regular_chain, regular_chain}\]

\[
\left[
\begin{array}{c}
  x - z = 0 \\
  y - z = 0 \\
  z^2 + 2z - 1 = 0
\end{array}
\right]
\quad
\left[
\begin{array}{c}
  x = 0 \\
  y = 0 \\
  z = 0
\end{array}
\right]
\quad
\left[
\begin{array}{c}
  x = 0 \\
  y - 1 = 0 \\
  z - 1 = 0
\end{array}
\right]
\quad
\left[
\begin{array}{c}
  x - 1 = 0 \\
  y = 0 \\
  z = 0
\end{array}
\right]
\]

\[\text{boxes} \coloneqq \text{seq}(\text{op}(\text{RealRootIsolate}(\text{rc}, R, '\text{rrr} = \frac{1}{29})), \text{rc} = \text{dec}); \quad \text{map}(\text{Display}, \text{boxes}, R)\]

\[\text{box, box, box, box, box}\]

\[
\left[
\begin{array}{c}
  x = \left[\frac{3393}{8192}, \frac{6791}{16384}\right] \\
  y = \left[\frac{3393}{8192}, \frac{6791}{16384}\right] \\
  z = \left[\frac{217167}{524288}, \frac{868669}{2097152}\right]
\end{array}\right]
\quad
\left[
\begin{array}{c}
  x = \left[\frac{-4947}{2048}, \frac{-2471}{1024}\right] \\
  y = \left[\frac{-4947}{2048}, \frac{-2471}{1024}\right] \\
  z = \left[\frac{-79109}{32768}, \frac{-316435}{131072}\right]
\end{array}\right]
\quad
\left[
\begin{array}{c}
  x = 0 \\
  y = 0 \\
  z = 1
\end{array}\right]
\quad
\left[
\begin{array}{c}
  x = 0 \\
  y = 1 \\
  z = 0
\end{array}\right]
\quad
\left[
\begin{array}{c}
  x = 0 \\
  y = 0 \\
  z = 0
\end{array}\right]
\]
Cylindrical algebraic decomposition of \( \{ax^2 + bx + c\} \)

The cylindrical algebraic decomposition of \( \{ax^2 + bx + c\} \) is given by the tree above, where \( t = bx + c \), \( q = 2ax + b \), and \( r = 4ac - b^2 \). This is the best possible output for that method, leading to 27 cells!
Can a computer program be as good as a high-school student?

For the equation $ax^2 + bx + c = 0$, can a computer program produce?

\[
\begin{align*}
&\begin{cases}
ax^2 + bx + c = 0 \\
a \neq 0 \land 4ac - b^2 > 0
\end{cases} \\
&\begin{cases}
2ax + b = 0 \\
4ac - b^2 = 0 \\
a \neq 0
\end{cases}
\end{align*}
\]

\[
\begin{align*}
&\begin{cases}
bx + c = 0 \\
a = 0 \\
b \neq 0
\end{cases} \\
&\begin{cases}
c = 0 \\
b = 0 \\
a = 0
\end{cases}
\end{align*}
\]
Yes, RealTriangularize in **MAPLE** can do that!

```maple
with(RegularChains); with(SemiAlgebraicSetTools); with(ParametricSystemTools); with(ParametricSystemTools);

R := PolynomialRing([x, c, b, a]); F := [a * x^2 + b * x + c];

polynomial_ring

\[ a x^2 + b x + c \]

Solving for the real solutions:
RealTriangularize(F, R, output = record);

\[
\begin{align*}
    a x^2 + b x + c &= 0 \\
    b x + c &= 0 \\
    c &= 0 \\
    -4 c a + b^2 &> 0 \\
    b &= 0 \\
    a &= 0 \\
    2 a x + b &= 0 \\
    a &
eq 0
\end{align*}
\]

Solving for the complex solutions

dec := Triangularize(F, R, output = lazard); map(Display, dec, R);

\[
\begin{align*}
    \text{regular_chain, regular_chain, regular_chain} \\
    \begin{align*}
        a x^2 + b x + c &= 0 \\
        a &
eq 0 \\
    \end{align*} \quad \begin{align*}
        b x + c &= 0 \\
        a &= 0 \\
    \end{align*} \quad \begin{align*}
        c &= 0 \\
    \end{align*} \quad \begin{align*}
        b &= 0 \\
        a &
eq 0 \\
    \end{align*} \quad \begin{align*}
        a &= 0
    \end{align*}
\]
```

\[ \text{polynomial_ring} \]

\[ a x^2 + b x + c \]
RealTriangularize applied to the \textit{Eve} surface (1/2)
RealTriangularize applied to the Eve surface (2/2)

\[ R := \text{PolynomialRing}([x, y, z]); F := [5x^2 + 2xz^2 + 5y^6 + 15y^4 + 5z^2 - 15y^5 - 5y^3, 5x^2 + 2xz^2 + 5y^6 + 15y^4 + 5z^2 - 15y^5 - 5y^3] \]

RealTriangularize(F, R, output = record);

\[
\begin{align*}
5x^2 + 2xz^2 + 5y^6 + 15y^4 - 5y^3 - 15y^5 + 5z^2 & = 0 \\
25y^6 - 75y^5 + 75y^4 - z^4 - 25y^3 + 25z^2 & < 0 \\
5x + z^2 & = 0 \\
25y^6 - 75y^5 + 75y^4 - 25y^3 - z^4 + 25z^2 & = 0 \\
64z^4 - 1600z^2 + 25 & > 0 \\
z & \neq 0 \\
z - 5 & \neq 0 \\
z + 5 & \neq 0 \\
x = 0 & , x = 0 & , x + 5 = 0 \\
y - 1 = 0 & , y - 1 = 0 & , y + 5 = 0 \\
z = 0 & , z = 0 & , z - 5 = 0 \\
5x + z^2 & = 0 \\
2y - 1 = 0 & \text{ } \\
64z^4 - 1600z^2 + 25 & = 0
\end{align*}
\]
Triangularize (not RealTriangularize) applied to sofa and cylinder (1/2)

\[ x^2 + y^3 + z^5 = x^4 + z^2 - 1 = 0 \]
Triangularize applied to sofa and cylinder (2/2)

```maple
> R := PolynomialRing([z, y, x]): F := [x^4+3x^2+y^3, x^4+y^3-1]: dec := Triangularize(F, R): map(Display, dec, R);

\[
\begin{pmatrix}
-2x^4 + x^8 + 1 \quad z + x^2 + y^3 = 0 \\
y^6 + 2x^2y^3 + 10x^{12} - 10x^8 + x^{20} - 5x^{16} + 6x^4 - 1 = 0 \\
-2x^4 + x^8 + 1 \neq 0
\end{pmatrix}
\]

> dec := Triangularize(F, R, output=lazard): map(Display, dec, R);

\[
\begin{pmatrix}
-2x^4 + x^8 + 1 \quad z + x^2 + y^3 = 0 \\
y^6 + 2x^2y^3 + 10x^{12} - 10x^8 + x^{20} - 5x^{16} + 6x^4 - 1 = 0 \\
-2x^4 + x^8 + 1 \neq 0
\end{pmatrix}
\]

\[
\begin{cases}
z = 0 \\
y^2 + y + 1 = 0 \\
x^2 + 1 = 0
\end{cases}
\]

\[
\begin{cases}
z = 0 \\
y - 1 = 0 \\
x^2 + 1 = 0
\end{cases}
\]

\[
\begin{cases}
z = 0 \\
y^2 - y + 1 = 0 \\
x + 1 = 0
\end{cases}
\]

\[
\begin{cases}
z = 0 \\
y^2 - y + 1 = 0 \\
x - 1 = 0
\end{cases}
\]

\[
\begin{cases}
z = 0 \\
y + 1 = 0 \\
x + 1 = 0
\end{cases}
\]

\[
\begin{cases}
z = 0 \\
y + 1 = 0 \\
x - 1 = 0
\end{cases}
\]
```
Solving for the integer solutions of a linear system (1/3)

Solve integer programming:

\[
\min_{\text{lex}}(x_1, \ldots, x_d) \\
Ax \leq b, \\
x \in \mathbb{Z}^d
\]

Example Problem:

\[
\begin{align*}
\min_{\text{lex}}(x_3, x_2, x_1) \\
3x_1 - 2x_2 + x_3 & \leq 7 \\
-2x_1 + 2x_2 - x_3 & \leq 12 \\
-4x_1 + x_2 + 3x_3 & \leq 15 \\
-x_2 & \leq -25 \\
x_1, x_2, x_3 & \in \mathbb{Z}
\end{align*}
\]
Example

Input: $K_1$ : \[
\begin{cases}
3x_1 - 2x_2 + x_3 \leq 7 \\
-2x_1 + 2x_2 - x_3 \leq 12 \\
-4x_1 + x_2 + 3x_3 \leq 15 \\
-x_2 \leq -25
\end{cases}, \quad \text{assume } x_1 > x_2 > x_3.
\]

Output: $K^1_1, K^2_1, K^3_1, K^4_1, K^5_1$ given by:

\[
\begin{cases}
3x_1 - 2x_2 + x_3 \leq 7 \\
-2x_1 + 2x_2 - x_3 \leq 12 \\
-4x_1 + x_2 + 3x_3 \leq 15 \\
2x_2 - x_3 \leq 48 \\
-5x_2 + 13x_3 \leq 67 \\
-x_2 \leq -25
\end{cases}, \quad \begin{cases}
x_1 = 15 \quad &x_1 = 18 \quad &x_1 = 14 \quad &x_1 = 19 \\
x_2 = 27 \quad &x_2 = 33 \quad &x_2 = 25 \quad &x_2 = 50 + t \\
x_3 = 16 \quad &x_3 = 18 \quad &x_3 = 15 \quad &x_3 = 50 + 2t \\
-25 \leq t \leq -16.
\end{cases}
\]
Solving for the integer solutions of a linear system (2/3)

\[
\min(x_3, x_2, x_1) \\
K_1 \cap \mathbb{Z}^3
\]

\[
\begin{align*}
\min(x_3, x_2, x_1) & \quad K_1^1 \cap \mathbb{Z}^3 \\
(2, -8, -4) & \\
\min(x_3, x_2, x_1) & \quad K_1^2 \cap \mathbb{Z}^3 \\
(16, 27, 15) & \\
\min(x_3, x_2, x_1) & \quad K_1^3 \cap \mathbb{Z}^3 \\
(18, 33, 18) & \\
\min(x_3, x_2, x_1) & \quad K_1^4 \cap \mathbb{Z}^3 \\
(15, 25, 14) & \\
\min(x_3, x_2, x_1) & \quad K_1^5 \cap \mathbb{Z}^3 \\
(0, 25, 19)
\end{align*}
\]
Plan

Polynomial system solvers in action

Algebraic structures

Polynomials in algebra

Univariate polynomial data-type

Multivariate polynomial data-type

Polynomial representations by values

Straighth-line programs

Big integers
Operation (1/2)

Definition
Given a non-empty set $\mathbb{M}$, an internal operation (or simply operation) over $\mathbb{M}$ is a function $f$ that maps any couple $(x, y)$ of elements from $\mathbb{M}$ with an element $f(x, y)$ of $\mathbb{M}$. The operation $f$

- is associative if the following holds
  \[ (\forall x, y, z \in \mathbb{M}) \quad f(x, f(y, z)) = f(f(x, y), z), \]

- is commutative if the following holds
  \[ (\forall x, y \in \mathbb{M}) \quad f(x, y) = f(y, x). \]

The set $\mathbb{M}$ possesses an identity element if there exists $e \in \mathbb{M}$ such that

\[ (\forall x \in \mathbb{M}) \quad f(e, x) = x = f(x, e) \]

Moreover, in this case, an element $x \in \mathbb{M}$ possesses a symmetric element (or reciprocal element) if the following holds

\[ (\exists x' \in \mathbb{M}) \quad f(x, x') = f(x', x) = e \]
Proposition
Let $M$ be a non-empty set with an operation $f$.

(i) If $M$ possesses an identity element, then it is unique.

(ii) Moreover, in this case, if an element $x \in M$ possesses a symmetric element $x' \in M$, then it is unique.

Remark
For a non-empty set $M$ with an associative operation $f$ it is natural to define $f(x_1, x_2, \ldots, x_n)$ for $x_1, x_2, \ldots, x_n \in M$ with $n \geq 3$ by

$$f(x_1, x_2, \ldots, x_n) = f(x_1, f(x_2, \ldots, x_n))$$
Semi-group, group

A *semi-group* is a set $\mathbb{M}$ endowed with an operation such that this operation is associative.

- If for this operation, the set $\mathbb{M}$ admits an identity element, then it is said to be a *monoid*. Furthermore, if for this operation every element possesses a symmetric element, then the monoid is said to be a *group*.
- If this operation is commutative, then it is usually denoted additively (provided that this does lead to confusion with another operation) and the semi-group is said *abelian* or *commutative*. Otherwise this operation is usually denoted multiplicatively.
- If $\mathbb{M}$ is an abelian semi-group and a monoid, then its identity element is denoted $0$ and $\mathbb{M}$ is said to be an *abelian monoid*.
- If $\mathbb{M}$ is a monoid which is not known to be commutative then its identity element is denoted $1$.
- If $\mathbb{M}$ is an abelian monoid and a group, then the symmetric element of an element $x \in \mathbb{M}$ is denoted $-x$ and called the *opposite* of $x$. Moreover, in this case, $\mathbb{M}$ is said to be an *abelian group*.
- If $\mathbb{M}$ is a group which is not known to be commutative then the symmetric element of an element $x \in \mathbb{M}$ is denoted $x^{-1}$ and called the *multiplicative inverse* of $x$ (or simply the *inverse* of $x$).
Semi-ring

A semi-ring is a set $\mathbb{A}$ endowed with two operations one being denoted additively and the other being denoted multiplicatively, called respectively the addition of $\mathbb{A}$ and the multiplication of $\mathbb{A}$ such that

(i) $\mathbb{A}$ is an abelian monoid for its addition,
(ii) $\mathbb{A}^*$ is a semi-group for its multiplication,
(iii) the multiplication of $\mathbb{A}$ is distributive w.r.t. its addition, which means that the following two conditions hold:
   - $(\forall x, y, z \in \mathbb{A}) \ x(y + z) = xy + xz$ (left-distributivity),
   - $(\forall x, y, z \in \mathbb{A}) \ (y + z)x = yx + zx$ (right-distributivity).

where $\mathbb{A}^* = \mathbb{A} \setminus \{0\}$. 
Ring

If $\mathbb{A}$ is an abelian group for its addition, then $\mathbb{A}$ is said to be a *ring*. From now on, we assume that $\mathbb{A}$ is a ring.

- If $\mathbb{A}^*$ is a monoid for its multiplication, then $\mathbb{A}$ is said to be a *ring with identity element*.
- If $\mathbb{A}^*$ is an abelian semi-group for its multiplication, then $\mathbb{A}$ is said to be a *commutative ring*.
- If $\mathbb{A}^*$ is an abelian monoid for its multiplication, then $\mathbb{A}$ is said to be a *commutative ring with identity element*.
- If $\mathbb{A}^*$ is a group for its multiplication, then $\mathbb{A}$ is said to be a *division ring* (or a *skew field*).
- If $\mathbb{A}^*$ is an abelian group for its multiplication, then $\mathbb{A}$ is said to be a *field*.
Some properties of rings (1/2)

Let $\mathbb{A}$ be a ring. For $x, y, z \in \mathbb{A}$ we have

$$x(y - z) + xz = x((y - z) + z) = xy \quad \text{and} \quad (y - z)x + zx = ((y - z) + z)x = yx$$

We deduce:

$$x(y - z) = xy - xz \quad \text{and} \quad (y - z)x = yx - zx.$$  \tag{1}$$

By setting $y = z$ we obtain

$$x \times 0 = 0 = 0 \times x.$$  \tag{2}$$

By setting $y = 0$ in Equation (1) we obtain

$$x \times (-z) = -(xz) \quad \text{and} \quad (-z)x = -(zx)$$  \tag{3}$$

which implies

$$(-x)(-z) = xz.$$  \tag{4}$$

Then, for every positive integer $n \in \mathbb{N}$ we deduce from Equation (4)

$$(-x)^n = (-1)^n x^n$$  \tag{5}$$
Let $\mathbb{A}$ be a commutative ring with identity element. Let $x \in \mathbb{A}$. Because of the rule $x^{n+m} = x^n x^m$ with $n, m$ positive integers, it is natural to define

$$x^0 = 1 \quad (6)$$

Then, one obtains the *Newton binomial formula* for every $x, y \in \mathbb{A}$

$$(x + y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k} \quad (7)$$
Examples

We illustrate the above definitions.

- The set of the natural integer numbers \( \mathbb{N} \) (endowed with its natural addition and multiplication) is a semi-ring but not a ring.

- The set of the integer numbers \( \mathbb{Z} \) is a commutative ring with identity element, but not a field.

- For \( p \in \mathbb{Z} \) with \( p \geq 2 \), the subset \( p\mathbb{Z} \) of \( \mathbb{Z} \) consisting of the multiples of \( p \) is a commutative ring, but not a commutative ring with identity element.

- For \( n \geq 2 \), the set \( M_{n,n}(\mathbb{Z}) \) of the square matrices of order \( n \) with integer coefficients, is a ring with identity element, but not a commutative ring.
Complex numbers

- Let \( \mathbb{F} \) be a field such that for every element \( x \in \mathbb{F} \) we have \( x^2 \neq -1 \).
- Then, the subset \( \text{Complex}(\mathbb{F}) \) of \( \mathcal{M}_{2,2}(\mathbb{F}) \) (the ring of square matrices with order 2 and coefficients in \( \mathbb{F} \)) consisting of the matrices of the form

\[
C(a, b) = \begin{pmatrix} a & b \\ -b & a \end{pmatrix}
\]

is a field (for the addition and the multiplication of \( \mathcal{M}_{2,2}(\mathbb{F}) \)), called the complex field of \( \mathbb{F} \).
- It is also a vector subspace of \( \mathcal{M}_{2,2}(\mathbb{F}) \) with dimension 2.
Let $\mathbb{F}$ be a field such that for all $x, y, z \in \mathbb{F}$ we have $x^2 + y^2 + z^2 \neq -1$.

Then, the subset $\text{Quaternion}(\mathbb{F})$ of $\mathcal{M}_{4,4}(\mathbb{F})$ (the ring of square matrices with order 4 and coefficients in $\mathbb{F}$) consisting of the matrices of the form

$$H(a, b, c, d) = \begin{pmatrix}
d & a & b & c \\
-a & d & -c & b \\
-b & c & d & -a \\
-c & -b & a & d
\end{pmatrix}$$

is a division ring, which is not a field, called the *quaternion ring* of $\mathbb{F}$.

It is also a vector subspace of $\mathcal{M}_{4,4}(\mathbb{F})$ with dimension 4.
Plan

Polynomial system solvers in action

Algebraic structures

**Polynomials in algebra**

Univariate polynomial data-type

Multivariate polynomial data-type

Polynomial representations by values

Straight-line programs

Big integers
Monoid Rings (1/6)

Notation
From now on, we consider a semi-group $\mathbb{M}$ whose operation is denoted multiplicatively and a ring $\mathbb{A}$ which may not be commutative and which may not have an identity element.

- Let $\underline{a} = (a_m)_{m \in \mathbb{M}}$ be a sequence of elements of $\mathbb{A}$ indexed by $\mathbb{M}$, that is a map from $\mathbb{M}$ to $\mathbb{A}$.
- For every $m \in \mathbb{M}$ the element $a_m$ of the sequence $\underline{a}$ is called the coefficient at $m$ of $\underline{a}$. The support of $\underline{a}$ is the subset of $\mathbb{M}$ defined by

$$\text{supp}(\underline{a}) = \{ m \in \mathbb{M} \mid a_m \neq 0 \}$$

(8)

The elements of $\text{supp}(\underline{a})$ are called the monomials of $\underline{a}$.
- The sequence $\underline{a}$ is a linear combination of elements from $\mathbb{M}$ with coefficients from $\mathbb{A}$ if its support is finite.
- The set of the linear combinations of elements from $\mathbb{M}$ with coefficients from $\mathbb{A}$ is denoted by $\mathbb{A}[\mathbb{M}]$ and called the monoid ring of $\mathbb{M}$ over $\mathbb{A}$.
- An element of $\mathbb{A}[\mathbb{M}]$ which has only one monomial is called a term.

In the case where $\mathbb{M}$ is a group, then $\mathbb{A}[\mathbb{M}]$ is said to be a group ring.
Monoid Rings (2/6)

- Let \( a, b \) be in \( A[M] \).
- The *sum* of \( a \) and \( b \) is the map \( s \) from \( M \) to \( A \) defined for every \( m \in M \) by
  \[
  s_m = a_m + b_m
  \]
  and denoted by \( a + b \).
- The *product* of \( a \) and \( b \) is the map \( p \) from \( M \) to \( A \) defined for every \( m \in M \) by
  \[
  p_m = \sum (m', m'') \in M \times M \quad a_{m'} b_{m''} \quad m' m'' = m
  \]
  and denoted by \( ab \).
Proposition
For \( a, b \) in \( A[M] \) the sum \( a + b \) and the product \( ab \) belong to \( A[M] \).

Proposition
The set \( A[M] \) endowed with the addition
\[
(a, b) \mapsto a + b
\]
and the multiplication
\[
(a, b) \mapsto ab
\]
is a ring.
Proposition

- Assume that $\mathbb{M}$ is a monoid with identity element $1_{\mathbb{M}}$ and that $\mathbb{A}$ is a ring with (multiplicative) identity element $1_{\mathbb{A}}$.
- Let $\mathbf{1}$ be the element of $\mathbb{A}[\mathbb{M}]$ with support $\{1_{\mathbb{M}}\}$ and with coefficient $1_{\mathbb{A}}$ at $1_{\mathbb{M}}$.
- Then, the ring $\mathbb{A}[\mathbb{M}]$ has $\mathbf{1}$ as (multiplicative) identity element.
**Definition**
Assume again that $M$ is a monoid with identity element $1_M$ and that $A$ is a ring with identity element $1_A$. Then, we define a map from $A$ to $A[M]$ by

$$a \mapsto a1_M$$

(9)

where $a1_M$ is the element of $A[M]$ whose support is $\{1_M\}$ and whose coefficient at $1_M$ is $a$. This map allows us to view $A$ as a subset of $A[M]$.

**Proposition**
With the hypothesis of the above definition, let us assume that $A$ is a commutative ring. Then, for every $a \in A$ and every $b \in A[M]$ we have

$$(a1_M) b = b (a1_M).$$
Remark
It follows from Proposition 5 that every element of $A$ commutes with every element of $A[M]$. However, commutativity of the multiplication in $A[M]$ requires also commutativity for $M$.

Proposition
Assume that $M$ is an abelian monoid and that $A$ is a commutative ring. Then, the ring $A[M]$ is commutative too.
The free abelian monoid

- Let $X$ be a set. The free monoid generated by $X$ is the set denoted by $X^*$ of all words (or finite sequences) over $X$ endowed with the concatenation as multiplication and with the empty word $\varepsilon$ as identity element.
- For later use, we define $X^+ = X^* \setminus \{\varepsilon\}$.
- We consider in $X^*$ the following equivalence relation: two words $w, w'$ over $X$ are equivalent if for every $x \in X$ the number of occurrences of $x$ is the same in both $w$ and $w'$.
- The set of the residue classes of this relation is an abelian monoid (for the multiplication induced by that of $X^*$) called the free abelian monoid generated by $X$. Let us denote it by $\mathbb{X}$. 
Multivariate polynomials (1/4)

- Let $m$ be any element of $X$. For any $x \in X$, the number of occurrences of $x$ in a representative of $m$ is called the degree of $m$ w.r.t. $x$ and is denoted by $\text{deg}(m, x)$.
- The total degree of $m$ is the sum of the numbers $\text{deg}(m, x)$ where $x$ runs over the elements of $X$ occurring in $m$.
- The ring $\mathbb{A}[X]$ is also denoted by $\mathbb{A}[X]$ and its elements are called multivariate polynomials in $X$ with coefficients in $\mathbb{A}$. If $X$ is a finite set $\{x_1, \ldots, x_p\}$ then
  - $\mathbb{A}[X]$ is also denoted by $\mathbb{A}[x_1, \ldots, x_p]$. Let $p \in \mathbb{A}[X]$ be non-zero.
  - For any $x \in X$, the maximum value of $\text{deg}(m, x)$ for $m \in \text{supp}(p)$ is the degree of $p$ w.r.t. $x$ and is denoted by $\text{deg}(p, x)$.
  - The maximum total degree of a monomial of $p$ is called the total degree of $p$.
Univariate polynomials

- Assume from now on that $X$ is a singleton $\{x\}$.
- Observe that the free monoid generated by $X$ is clearly identical to the free abelian monoid generated by $X$.
- Moreover, every element of $\mathbb{A}[x]$ is called a univariate polynomial in $x$ with coefficients in $\mathbb{A}$.
- In addition, the total degree of a non-zero element $p$ of $\mathbb{A}[x]$ is simply called its degree and is denoted by $\text{deg}(p)$. 
Because the monoid ring $\mathbb{A}[M]$ is a generalisation of the polynomial ring $\mathbb{A}[x]$, it is natural and convenient to use the following notation. An element $a = (a_m)_{m \in M}$ of $\mathbb{A}[M]$ can be written

$$a = \sum_{m \in M} a_m$$
Without any additional assumption on \( \mathbb{M} \), computing in \( \mathbb{A}[\mathbb{M}] \) is not easy. First, one would like to have a *canonical way* to represent the elements of \( \mathbb{A}[\mathbb{M}] \). That would make the comparison or the addition of two elements from \( \mathbb{A}[\mathbb{M}] \) simpler. Second, computing the product of two elements \( a \) and \( b \) of \( \mathbb{A}[\mathbb{M}] \) implies to compute all the couples \((m', m'') \in \mathbb{M} \times \mathbb{M}\) such that \( m' m'' \) is equal to a given \( m \in \mathbb{M} \). If \( \mathbb{M} \) is a group, then the equation \( m' m'' = m \) is simpler since we must have \( m'' = m'^{-1} m \).

**definition**

A total order \( \leq \) on an abelian monoid \( \mathbb{M} \) is a *term order* if the following two conditions hold

(i) for every \( m \in \mathbb{M} \) we have \( 1_{\mathbb{M}} \leq m \)

(ii) for every \( m, m', m'' \in \mathbb{M} \) we have \( m \leq m' \Rightarrow mm'' \leq m'm'' \)
Assume that $\mathbb{M}$ is an abelian monoid endowed with a term order $\leq$ and let $a \in \mathbb{A}[\mathbb{M}]$ be a non-zero element.

- The maximum (w.r.t. the total order of $\mathbb{M}$) element of $\text{supp}(a)$ is called the *leading monomial of $a$* and is denoted by $\text{lm}(a)$.
- The coefficient of $a$ at $\text{lm}(a)$ is called the *leading coefficient of $a$* and is denoted by $\text{lc}(a)$.
- The term of $\mathbb{A}[\mathbb{M}]$ whose leading monomial is $\text{lm}(a)$ and whose leading coefficient is $\text{lc}(a)$ is called the *leading term of $a$* and is denoted by $\text{lt}(a)$.
- The element $a - \text{lt}(a)$ is called the *reductum of $a$*.
- Finally, the leading coefficient, the leading term and the reductum of $0$ are defined to be $0$.
- It is sometimes convenient to set $\text{lm}(0) = 0$ as well.
Example (1/4)

- This example is taken from Automata Theory and assume that the reader is familiar with the notion of a finite automaton.
- Let us consider an alphabet $\Sigma$, a finite automaton (not necessarily deterministic) $A$ recognising a language $L$ over $\Sigma$ and a positive integer $n$.
- We are interested in computing the words of $L$ with length $n$. 
Let $Q = \{1, \ldots, q\}$ be the set of states of $A$ and let $\Sigma^*$ be the set of words over $\Sigma$.

Recall that $\Sigma^*$ is a monoid whose identity element is the empty word.

Let $A$ be the ring $\mathbb{Z}[\Sigma^*]$ of linear combinations of words from $\Sigma^*$ with coefficients from the ring of integer numbers $\mathbb{Z}$.

Let $\delta : (Q, \Sigma \cup \{\varepsilon\}) \rightarrow 2^Q$ be the transition function of $A$. To every couple $(i, j) \in Q \times Q$ of states we associate the element $T_{i,j}$ of $\mathbb{Z}[\Sigma^*]$ defined by

$$ T_{i,j} = \sum_{\substack{x \in \Sigma \cup \{\varepsilon\} \\
j \in \delta(i, x)}} x. $$

In broad words, the element $T_{i,j}$ is the sum of the $x \in \Sigma \cup \{\varepsilon\}$ such that one transits from state $i$ to state $j$ by reading $x$. 
Let $T$ be the square matrix of order $q$ with coefficients in $\mathbb{Z}[\Sigma^*]$ such that $T_{i,j}$ is the element of $T$ at the intersection of row $i$ and column $j$.

Let $S$ be the horizontal vector of length $q$ with coefficients in $\mathbb{Z}$ such that $S_i = 1$ if $i$ is an initial state and $S_i = 0$ otherwise.

Let $F$ be the vertical vector of length $q$ with coefficients in $\mathbb{Z}$ such that $F_i = 1$ if $i$ is a final state and $F_i = 0$ otherwise.

Then we define the following element of $\mathbb{Z}[\Sigma^*]$

$$p_n(\mathcal{A}) = ST^nF.$$ 

Let us compute this quantity for $n = 2$, $\Sigma = \{a, b\}$, $Q = \{1, 2\}$, the initial state $1$, the final state $2$ and the following transition function

<table>
<thead>
<tr>
<th></th>
<th>$a$</th>
<th>$b$</th>
<th>$\varepsilon$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1,2</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>$\emptyset$</td>
</tr>
</tbody>
</table>
Example (4/4)

Then, the matrix $T$ is

$$T = \begin{pmatrix} a + b & b \\ 0 & a + b \end{pmatrix}$$

and its square is

$$T^2 = \begin{pmatrix} (a + b)(a + b) & (a + b)b + b(a + b) \\ 0 & (a + b)(a + b) \end{pmatrix} = \begin{pmatrix} a^2 + ab + ba + b^2 & ab + ba + 2b^2 \\ 0 & a^2 + ab + ba + b^2 \end{pmatrix}$$

Then we have

$$p_2(A) = ST^2F = \begin{pmatrix} 1 & 0 \end{pmatrix} T^2 \begin{pmatrix} 0 \\ 1 \end{pmatrix} = ab + ba + 2b^2$$

Observe that $ab$, $ba$ and $b^2$ are the three words of length 2 that the automaton $A$ recognises. The coefficient 2 of $b^2$ comes from the fact that there are two ways to recognise this word.
Plan

Polynomial system solvers in action

Algebraic structures

Polynomials in algebra

Univariate polynomial data-type

Multivariate polynomial data-type

Polynomial representations by values

Straigth-line programs

Big integers
Univariate polynomial data-type

Let $\mathbb{A}$ be ring. The univariate polynomial of $\mathbb{A}[x]$ can be implemented using different data-types in a computer program:

- dense univariate polynomial (DUP)
- sparse univariate polynomial (SUP)
- Straight-line program (SLP)
- ...
Dense univariate polynomial (DUP)

The polynomial

\[ p(x) = a_n x^n + \cdots + a_1 x + a_0 \]  \hspace{1cm} (10)

is coded by a record consisting of

- a single integer \( s \),
- a single integer \( d \leq s + 1 \),
- an array of size \( s \) such that \( a_0 + \cdots + a_n x^n \) is represented by \([a_0, \ldots, a_n, \ldots]\) and \( d = n \).

- This representation is said *dense* because all \( a_i \) are coded, even those which are null.
- This representation is said *canonical* because two different polynomials have different such representations.
- Hence operations like \texttt{DEGREE}, \texttt{LEADING COEFFICIENT}, \texttt{REDUCTUM} are in \( \mathcal{O}(1) \).
- Addition and equality-test are in \( \mathcal{O}(n) \) and multiplication is in \( \mathcal{O}(n^2) \).
- This representation is especially good when the ring of coefficients is a small prime field, i.e. \( \mathbb{Z}/p\mathbb{Z} \) with \( p \) prime and in the range \([2, 2^N - 1]\), for a fixed \( N \).
Sparse univariate polynomial (SUP)

The polynomial

\[ p(x) = a_n x^n + \cdots + a_1 x + a_0 \quad (11) \]

is coded by the list \( L \) of records \([a_i, i]\) where \( a_i \) is a nonzero coefficient and such that \( L \) is sorted decreasingly w.r.t. \( i \).

- This representation is said *sparse*, since only the nonzero \( a_i \) are coded.
- This representation is also canonical.
- Hence operations like \textsc{degree}, \textsc{leading coefficient}, \textsc{reductum} are in \( \mathcal{O}(1) \).
- Moreover the operation \textsc{reductum} does not require coefficient duplication (on the contrary of the previous representation).
- Addition and equality-test are in \( \mathcal{O}(n) \) and multiplication is in \( \mathcal{O}(n^2) \).
- This representation is especially good when the ring of coefficients is itself a ring of sparse polynomials.
Division with remainder

**Input:** univariate polynomials \( f = \sum_{0}^{n} a_i x^i \) and \( g = \sum_{0}^{m} b_i x^i \) in \( \mathbb{A}[x] \) with respective degrees \( n \) and \( m \) such that \( b_m \) is a unit.

**Output:** the quotient \( q \) and the remainder \( r \) of \( f \) w.r.t. \( g \).

\[
\begin{align*}
n < m & \Rightarrow \textbf{return } (0, f) \\
r & := f \\
\textbf{for } i = n - m, n - m - 1, \ldots, 0 & \textbf{ repeat} \\
\textbf{if} \ \deg r = m + i & \textbf{ then} \\
\quad q_i & := \text{lcl}(r) / b_m \\
\quad r & := r - q_i x^i g \\
\textbf{else} & q_i := 0 \\
q & := \sum_{n-m}^{0} q_i x^i \\
\textbf{return } (q, r)
\end{align*}
\]

**Exercise**
Assuming that each element of \( \mathbb{A} \) can fit a machine word, what is the minimum space requirement for implementing the above algorithm in the case of DUP? SUP?
Plan

Polynomial system solvers in action

Algebraic structures

Polynomials in algebra

Univariate polynomial data-type

Multivariate polynomial data-type

Polynomial representations by values

Straigth-line programs

Big integers
Univariate polynomial data-type

Let again $\mathbb{A}$ be ring and let $X = \{x_1, \ldots, x_n\}$ be a finite set of variables. The univariate polynomial of $\mathbb{A}[X] = \mathbb{A}[X]$ (hence in commutative variables) can be implemented using different data-types in a computer program:

- recursively based on SUP
- recursively based on DUP
- Expanded (or distributed) multivariate data-type
- Straight-line program (SLP)
- ...
Recursively

Recall $X = \{x_1, \ldots, x_n\}$ and we want to implement $\mathbb{A}[X]$.

- If $n = 1$ we can use a univariate representation
- otherwise we can view $\mathbb{A}[X]$ as a univariate polynomial ring with a multivariate polynomial ring as coefficient ring, say for instance $\mathbb{A}[x_1, \ldots, x_{n-1}][x_n]$.

This representation

- implies to choose an ordering on the variables and a representation for univariate polynomials.
- is well adapted for certain operations, in particular those around the notion of GCD (Greatest Common Divisor).
- More on this later.
Each polynomial can be viewed as a linear combination of monomials (with coefficients in $R$).

Then the polynomial

$$p = a_1 m_1 + \cdots + a_t m_t.$$  \hspace{1cm} (12)

where the $m_i$ are pairwise different monomials and the $a_i$ are nonzero coefficients, can be represented as an aggregate of terms $[a_i, m_i]$.

Once an order is chosen on $X$, say $x_1 > x_2 > \cdots > x_n$, a monomial $x_1^{e_1} x_2^{e_2} \cdots x_n^{e_n}$ is generally by the exponent vector $[e_1, e_2, \ldots, e_n]$.

Assume that we monomials are totally ordered Assume also that the aggregate is linear, that is, defining a 1-to-1 map from $[1, t] \cap \mathbb{N}$ to the terms of $p$ should be a first term, a second term, \ldots Finally, assume that this map sorts terms decreasingly. Then, this provides us with a canonical representation for $\mathbb{A}[X]$.

The most commonly used aggregate is linked list.

One can also consider alternating arrays (thus alternating one coefficient and one monomial).
Two types of monomial orderings are frequently used.

- The lexicographical ordering. With \( X = \{ x > y > z \} \) we have
  \[
  1 < z < \cdots < z^n < y < yz < \cdots < yz^n < y^2 < y^2 z < \cdots < y^2 z^n \quad (13)
  \]

- The degree-lexicographical ordering. With \( X = \{ x > y > z \} \) we have
  \[
  1 < z < y < x < z^2 < zy < y^2 < zx < xy < x^2 < \cdots < \quad (14)
  \]
Sparse multivariate addition (1/2)

Compute $C := A + B$ where

- $A = \sum_{i=1}^{n} a_i x^{\alpha_i}$,
- $B = \sum_{j=1}^{m} b_j x^{\beta_j}$,
- $C = \sum_{k=1}^{K} c_k x^{\gamma_k}$,

where

- $\alpha_i, \beta_j, \gamma_k$ are exponent vectors,
- $a_i, b_j, c_k$ are coefficients,
- $m, n, K$ are the number of terms of $A, B, C$ respectively,
- terms are sorted decreasingly in the above expressions.

The algorithm on the right-hand side performs $O(m + n)$ comparisons of exponent vectors and additions in $A$. 
Sparse multivariate addition (1/2)

Compute \( C := A + B \) where

\( A = \sum_{i=1}^{n} a_i x^{\alpha_i} \),

\( B = \sum_{j=1}^{m} b_j x^{\beta_j} \),

\( C = \sum_{k=1}^{K} c_k x^{\gamma_k} \),

where

\( \alpha_i, \beta_j, \gamma_k \) are exponent vectors,

\( a_i, b_j, c_k \) are coefficients,

\( m, n, K \) are the number of terms of \( A, B, C \) respectively,

terms are sorted decreasingly in the above expressions.

The algorithm on the right-hand side performs \( \mathcal{O}(m + n) \) comparisons of exponent vectors and additions in \( A \).
\( \kappa = 0 \)
\( i = 1 \)
\( j = 1 \)
\[ \text{while ( } i \leq n \text{ and } j \leq m \text{ )} \]
\[ k = k + 1 \]
\[ \text{if ( } \alpha_i < \beta_j \text{ )} \]
\[ c_k = b_j \]
\[ \gamma_k = \beta_j \]
\[ j = j + 1 \]
\[ \text{else if ( } \alpha_i = \beta_j \text{ )} \]
\[ c_k = a_i + b_j \]
\[ \gamma_k = \alpha_i \]
\[ \text{if ( } c_k = 0 \text{ ) } k = k - 1 \]
\[ i = i + 1 \]
\[ j = j + 1 \]
\[ \text{else if ( } \alpha_i > \beta_j \text{ )} \]
\[ c_k = a_i \]
\[ \gamma_k = \alpha_i \]
\[ i = i + 1 \]
\[ \text{while ( } i \leq n \text{ )} \]
\[ k = k + 1 \]
\[ c_k = a_i \]
\[ \gamma_k = \alpha_i \]
\[ i = i + 1 \]
\[ \text{while ( } j \leq m \text{ )} \]
\[ k = k + 1 \]
\[ c_k = b_j \]
\[ \gamma_k = \beta_j \]
\[ j = j + 1 \]
\[ K = k \]
Sparse multivariate multiplication (1/4)

Now, we want to compute $C := AB$ with, as above,

$$A = \sum_{i=1}^{i=n} a_i x^{\alpha_i}, \quad B = \sum_{j=1}^{j=m} b_j x^{\beta_j}, \quad \text{and} \quad C = \sum_{k=1}^{k=K} c_k x^{\gamma_k}. \quad (15)$$

What is the cost of a *plain multiplication*?

- Generating all terms costs $O(nm)$,
- Sorting them all costs $O(nm \log(nm))$,
- Combining terms of all equal exponent vectors $O(nm)$.
- This yields an arithmetic complexity of $O(nm \log(nm))$,
- with a space complexity of $\Theta(nm)$, which can be improved.
Sparse multivariate multiplication (2/4)

Using distributivity naively

- Write $C = \sum_{i=1}^{i=n} \left( \sum_{j=1}^{j=m} a_i b_j x^{\alpha_i + \beta_j} \right)$
- Suppose we add the $n$ summands, one after another.
- Then the $i$-th summation may cost $\mathcal{O}(im + n)$ arithmetic operations
- This yields a total arithmetic complexity of $\mathcal{O}(n^2 m)$, but reduces space complexity when cancellations happens

Divide-and-conquer approach

- Instead of summing the summands, one after another, proceed in a divide-and-conquer manner
- Assume $m \leq n$ and let $C(n)$ be the cost of adding $n$ polynomials of size $m$. We have

$$C(n) = 2C(n/2) + \left( \frac{mn}{2} + \frac{mn}{2} \text{right} \right)$$

(16)

- This yields $C(n) \in \mathcal{O}(nm \log(nm))$. 
Sparse multivariate multiplication (3/4)

- Consider now algorithms attempting to generate terms by decreasing order of exponent vectors.
- The term with exponent $\alpha_i + \beta_j$ appears in the product before the term with exponent $\alpha_i + \beta_{j+1}$.
- Thus, at each step of those algorithms, and for each $1 \leq i \leq n$, there exists an index $f_i$ such that terms with exponent $\alpha_i + \beta_j$ have (resp. have not) been included in the answer for $j < f_i$ (resp. $j \leq f_i$).
- The exponent of the next term to be included in the answer will be the largest of the $\alpha_i + \beta_{f_i}$ where $i$ ranges from 1 to $n$.
- The $f_i$ are decreasing with $i$; thus if $f_i > m$ for some index $i$, then $f_j > m$ holds for all $j \leq i$.
- In the algorithm, $I$ is the smallest $i$ such that $f_i \leq m$ holds.
Sparse multivariate multiplication (3/4)

- Consider now algorithms attempting to generate terms by decreasing order of exponent vectors.
- The term with exponent $\alpha_i + \beta_j$ appears in the product before the term with exponent $\alpha_i + \beta_{j+1}$
- Thus, at each step of those algorithms, and for each $1 \leq i \leq n$, there exists an index $f_i$ such that terms with exponent $\alpha_i + \beta_j$ have (resp. have not) been included in the answer for $j < f_i$ (resp. $j \leq f_i$).
- The exponent of the next term to be included in the answer will be the largest of the $\alpha_i + \beta_{f_i}$ where $i$ ranges from 1 to $n$.
- The $f_i$ are decreasing with $i$; thus if $f_i > m$ for some index $i$, then $f_j > m$ holds for all $j \leq i$
- In the algorithm, $l$ is the smallest $i$ such that $f_i \leq m$ holds.

```plaintext
if (m=0 or n=0) {
    K=0
    return
}
k=1
c=0
\gamma_1 = \alpha_1 + \beta_1
for i=1 to n do f_i = 1
l=1
while (l \leq n) {
    if (c_k \neq 0) {
        k=k+1
        c_k=0
        \gamma_k = \alpha_s + \beta_{f_s}
        f_s = f_s+1
        if (f_s > m) l=l+1
    }
k=k
```

```plaintext
K=k
```
This algorithm runs in $O(mn \log(n))$. Explain why! What is its space complexity?
Plan

Polynomial system solvers in action

Algebraic structures

Polynomials in algebra

Univariate polynomial data-type

Multivariate polynomial data-type

Polynomial representations by values

Straigth-line programs

Big integers
Polynomials and the Fast Fourier Transform (FFT)

- http://web.cecs.pdx.edu/~maier/cs584/Lectures/lect07b-11-MG.pdf
Plan

Polynomial system solvers in action

Algebraic structures

Polynomials in algebra

Univariate polynomial data-type

Multivariate polynomial data-type

Polynomial representations by values

Straight-line programs

Big integers
- Horner’s method
  https://en.wikipedia.org/wiki/Horner%27s_method
- SLP in Wikopedia
  https://en.wikipedia.org/wiki/Straight-line_program
- SLP in Chapter 2 of http://www.csd.uwo.ca/~moreno/
  /Publications/Liyun.Li-MasterThesis-2010.pdf
Plan

Polynomial system solvers in action

Algebraic structures

Polynomials in algebra

Univariate polynomial data-type

Multivariate polynomial data-type

Polynomial representations by values

Straight-line programs

Big integers
Course notes:
  - https://people.eecs.berkeley.edu/~fateman/282/F%20Wright%20notes/week4.pdf

Additional course notes:

Demo codes:
  - http://faculty.cse.tamu.edu/djimenez/ut/utsa/cs3343/lecture20.html
  - https://www3.cs.stonybrook.edu/~skiena/392/programs/bignum.c

Professional codes:
  - GMP https://gmplib.org/
  - NTL http://www.flintlib.org/
  - FLINT http://www.shoup.net/ntl/