Triangular Decomposition of Polynomial Systems: Algorithmic Advances and Remaining Challenges

Marc Moreno Maza
(Univ. of Western Ontario)

ICMM'09
In honor of Prof. Wen-Tsün Wu
Beijing, China,
May 13, 2009

Wu's Characteristic Set Method

Input: $F \subset \mathbf{k}[x_1, \ldots, x_n]$ and a **variable ordering** $\leq$.

Output: C a Wu characteristic set of F.

repeat

(S) $B := \mathrm{MinimalAutoreducedSubset}(F, \leq)$

(R) $A := F \setminus B$;

$R :=$ **prem** (A, B)

(U) $R := R \setminus \{0\}$; $F := F \cup R$

until $R = \emptyset$

return B

- Repeated calls decomposes $V(F)$ into quasi-components.

Buchberger's Algorithm

Input: $F \subset \mathbf{k}[x_1, \ldots, x_n]$ and a **term order** $\leq$.

Output: G a reduced Gröbner basis w.r.t. $\leq$ of the ideal $\langle F \rangle$ generated by F.

repeat
(S) $B := \mathrm{MinimalAutoreducedSubset}(F, \leq)$
(R) $A :=$ **S_Polynomials** $(B) \cup F$;
$R :=$ **Reduce** $(A, B, \leq)$
(U) $R := R \setminus \{0\}$; $F := F \cup R$
until $R = \emptyset$
return B

The Notion of a Regular Chain (1/2)

- Let $T \subset \mathbf{k}[x_1 < \cdots < x_n] \setminus \mathbf{k}$ be a triangular set, hence the polynomials of T have pairwise distinct main variables.

The Notion of a Regular Chain (1/2)

- Let $T \subset \mathbf{k}[x_1 < \cdots < x_n] \setminus \mathbf{k}$ be a triangular set, hence the polynomials of T have pairwise distinct main variables.

- Let $\mathrm{mvar}(T) := \{\mathrm{mvar}(t) \ \mid \ t \in T\}$, $\mathrm{init}(t) := \mathrm{lc}(t, \mathrm{mvar}(t))$ for all $t \in T$, and $h_T := \prod_{t \in T} \mathrm{init}(t)$.

The Notion of a Regular Chain (1/2)

- Let $T \subset \mathbf{k}[x_1 < \cdots < x_n] \setminus \mathbf{k}$ be a triangular set, hence the polynomials of T have pairwise distinct main variables.
- Let $\mathrm{mvar}(T) := \{\mathrm{mvar}(t) \mid t \in T\}$, $\mathrm{init}(t) := \mathrm{lc}(t, \mathrm{mvar}(t))$ for all $t \in T$, and $h_T := \prod_{t \in T} \mathrm{init}(t)$.
- T_v is the polynomial of T with main variable v, for $v \in \mathrm{mvar}(T)$, and $T_{<v} := \{t \in T \mid \mathrm{mvar}(t) < v\}$.

The Notion of a Regular Chain (1/2)

- Let $T \subset \mathbf{k}[x_1 < \cdots < x_n] \setminus \mathbf{k}$ be a triangular set, hence the polynomials of T have pairwise distinct main variables.
- Let $\mathrm{mvar}(T) := \{\mathrm{mvar}(t) \mid t \in T\}$, $\mathrm{init}(t) := \mathrm{lc}(t, \mathrm{mvar}(t))$ for all $t \in T$, and $h_T := \prod_{t \in T} \mathrm{init}(t)$.
- T_v is the polynomial of T with main variable v, for $v \in \mathrm{mvar}(T)$, and $T_{<v} := \{t \in T \mid \mathrm{mvar}(t) < v\}$.
- The quasi-component of T is $W(T) = V(T) \setminus V(h_T)$.

The Notion of a Regular Chain (1/2)

- Let $T \subset \mathbf{k}[x_1 < \cdots < x_n] \setminus \mathbf{k}$ be a triangular set, hence the polynomials of T have pairwise distinct main variables.
- Let $\mathrm{mvar}(T) := \{\mathrm{mvar}(t) \mid t \in T\}$, $\mathrm{init}(t) := \mathrm{lc}(t, \mathrm{mvar}(t))$ for all $t \in T$, and $h_T := \prod_{t \in T} \mathrm{init}(t)$.
- T_v is the polynomial of T with main variable v, for $v \in \mathrm{mvar}(T)$, and $T_{<v} := \{t \in T \mid \mathrm{mvar}(t) < v\}$.
- The quasi-component of T is $W(T) = V(T) \setminus V(h_T)$.
- The saturated ideal of T is the ideal of $\mathbf{k}[x_1 < \cdots < x_n]$

$$\mathrm{sat}(T) := \langle T \rangle : (h_T)^\infty.$$

The Notion of a Regular Chain (1/2)

- Let $T \subset \mathbf{k}[x_1 < \cdots < x_n] \setminus \mathbf{k}$ be a triangular set, hence the polynomials of T have pairwise distinct main variables.
- Let $\mathrm{mvar}(T) := \{\mathrm{mvar}(t) \mid t \in T\}$, $\mathrm{init}(t) := \mathrm{lc}(t, \mathrm{mvar}(t))$ for all $t \in T$, and $h_T := \prod_{t \in T} \mathrm{init}(t)$.
- T_v is the polynomial of T with main variable v, for $v \in \mathrm{mvar}(T)$, and $T_{<v} := \{t \in T \mid \mathrm{mvar}(t) < v\}$.
- The quasi-component of T is $W(T) = V(T) \setminus V(h_T)$.
- The saturated ideal of T is the ideal of $\mathbf{k}[x_1 < \cdots < x_n]$

$$\mathrm{sat}(T) := \langle T \rangle : (h_T)^{\infty}.$$

- T is a regular chain if for each $v \in \mathrm{mvar}(T)$ the initial of T_v is regular modulo $\mathrm{sat}(T_{<v})$ (Michael Kalkbrener 91).

The Notion of a Regular Chain (2/2)

- Let $p \in \mathbf{k}[x_1 < \cdots < x_n]$ and $T := T_{<w} \cup T_w \subset \mathbf{k}[x_1 < \cdots < x_n]$ be a triangular set. The *iterated resultant* of p w.r.t. T is:

$$\mathrm{res}(p, T) \quad = \quad \begin{cases} p & \text{if } \deg(f, w) = 0 \\ \mathrm{res}(\mathrm{res}(p, T_w, w), T_{<w}) & \text{otherwise} \end{cases}$$

The Notion of a Regular Chain (2/2)

- Let $p \in \mathbf{k}[x_1 < \cdots < x_n]$ and $T := T_{<w} \cup T_w \subset \mathbf{k}[x_1 < \cdots < x_n]$ be a triangular set. The *iterated resultant* of p w.r.t. T is:

$$\mathsf{res}(p, T) \quad = \quad \begin{cases} p & \text{if } \deg(f, w) = 0 \\ \mathsf{res}(\mathsf{res}(p, T_w, w), T_{<w}) & \text{otherwise} \end{cases}$$

- T is a regular chain iff

$$\mathsf{res}(h_T, T) \neq 0$$

(Lu Yang, Jingzhong Zhang 91).

The Notion of a Regular Chain (2/2)

- Let $p \in \mathbf{k}[x_1 < \cdots < x_n]$ and $T := T_{<w} \cup T_w \subset \mathbf{k}[x_1 < \cdots < x_n]$ be a triangular set. The *iterated resultant* of p w.r.t. T is:

$$\mathrm{res}(p, T) \quad = \quad \begin{cases} p & \text{if } \deg(f, w) = 0 \\ \mathrm{res}(\mathrm{res}(p, T_w, w), T_{<w}) & \text{otherwise} \end{cases}$$

- T is a regular chain iff

$$\mathrm{res}(h_T, T) \neq 0$$

(Lu Yang, Jingzhong Zhang 91).

- T is a regular chain iff

$$\{p \mid \mathrm{prem}(p, T) = 0\} \quad = \quad \mathrm{sat}(T)$$

(Philippe Aubry, Daniel Lazard, Marc Moreno Maza 97).

Regular Chain under Specialization

- Let $\mathbf{u} = u_1, \ldots, u_d$ be parameters, $\mathbf{y} = y_1, \ldots, y_m$ be unknowns, Π_U be the projection from $\mathbf{K}^{m+d}$ to $\mathbf{K}^d$.

Regular Chain under Specialization

- Let $\mathbf{u} = u_1, \ldots, u_d$ be parameters, $\mathbf{y} = y_1, \ldots, y_m$ be unknowns, Π_U be the projection from $\mathbf{K}^{m+d}$ to $\mathbf{K}^d$.
- A regular chain $T \subset \mathbf{k}[\mathbf{u}, \mathbf{y}]$ *specializes well* at $u \in \mathbf{K}^d$ if $T(u)$ is a regular chain in $\mathbf{K}[\mathbf{y}]$ and $\mathrm{rank}(T(u)) = \mathrm{rank}(T_{>U_d})$.

Regular Chain under Specialization

- Let $\mathbf{u} = u_1, \ldots, u_d$ be parameters, $\mathbf{y} = y_1, \ldots, y_m$ be unknowns, Π_U be the projection from $\mathbf{K}^{m+d}$ to $\mathbf{K}^d$.
- A regular chain $T \subset \mathbf{k}[\mathbf{u}, \mathbf{y}]$ *specializes well* at $u \in \mathbf{K}^d$ if $T(u)$ is a regular chain in $\mathbf{K}[\mathbf{y}]$ and $\mathrm{rank}(T(u)) = \mathrm{rank}(T_{>U_d})$.
- Let $T \subset \mathbf{k}[\mathbf{u}, \mathbf{y}]$ be a reg. chain and $u \in \Pi_U(\mathbf{W}(T \cap \mathbf{k}[U]))$. T specializes well at $u \iff \mathrm{res}(h_{T_{>U_d}}, T_{>U_d}) \neq 0$ at $\mathbf{u} = u$

Regular Chain under Specialization

- Let $\mathbf{u} = u_1, \ldots, u_d$ be parameters, $\mathbf{y} = y_1, \ldots, y_m$ be unknowns, Π_U be the projection from $\mathbf{K}^{m+d}$ to $\mathbf{K}^d$.
- A regular chain $T \subset \mathbf{k}[\mathbf{u}, \mathbf{y}]$ *specializes well* at $u \in \mathbf{K}^d$ if $T(u)$ is a regular chain in $\mathbf{K}[\mathbf{y}]$ and $\mathrm{rank}(T(u)) = \mathrm{rank}(T_{>U_d})$.
- Let $T \subset \mathbf{k}[\mathbf{u}, \mathbf{y}]$ be a reg. chain and $u \in \Pi_U(\mathbf{W}(T \cap \mathbf{k}[U]))$. T specializes well at $u \iff \mathrm{res}(h_{T_{>U_d}}, T_{>U_d}) \neq 0$ at $\mathbf{u} = u$
- Replacing *regular chain* by squarefree reg. ch. in char. 0 and $h_{T_{>U_d}}$ by $\mathrm{Sep}_{T_{>U_d}}$ one obtains the *border polynomial of T*.

Regular Chain under Specialization

- Let $\mathbf{u} = u_1, \ldots, u_d$ be parameters, $\mathbf{y} = y_1, \ldots, y_m$ be unknowns, Π_U be the projection from $\mathbf{K}^{m+d}$ to $\mathbf{K}^d$.
- A regular chain $T \subset \mathbf{k}[\mathbf{u}, \mathbf{y}]$ *specializes well* at $u \in \mathbf{K}^d$ if $T(u)$ is a regular chain in $\mathbf{K}[\mathbf{y}]$ and $\mathrm{rank}(T(u)) = \mathrm{rank}(T_{>U_d})$.
- Let $T \subset \mathbf{k}[\mathbf{u}, \mathbf{y}]$ be a reg. chain and $u \in \Pi_U(\mathbf{W}(T \cap \mathbf{k}[U]))$. T specializes well at $u \iff \mathrm{res}(h_{T_{>U_d}}, T_{>U_d}) \neq 0$ at $\mathbf{u} = u$
- Replacing *regular chain* by squarefree reg. ch. in char. 0 and $h_{T_{>U_d}}$ by $\mathrm{Sep}_{T_{>U_d}}$ one obtains the *border polynomial of T*.

Related Work

On a projection theorem of quasi-varieties in elimination theory (Wen-Tsün Wu 90). (Xiao-Shan Gao, Shang-Ching Chou 92) (Dongming Wang 00 & 01) (Lu Yang, Xiaorong Hou, Bican Xia 01) (Xiao-Shan Gao, Ding-Kang Wang 03) (Changbo Chen, Oleg Golubitsky, François Lemaire, Marc Moreno Maza, Wei Pan 07)

Incremental Solving

- Let $F \subset \mathbf{k}[\mathbf{x}]$, $f \in \mathbf{k}[\mathbf{x}]$, $T, T^m \ldots, T^e \subset \mathbf{k}[\mathbf{x}]$ reg. chains. Assume we have *solved* F as $V(F) = W(T^i) \cup \cdots \cup W(T^e)$.

Incremental Solving

- Let $F \subset \mathbf{k}[\mathbf{x}]$, $f \in \mathbf{k}[\mathbf{x}]$, $T, T^m \ldots, T^e \subset \mathbf{k}[\mathbf{x}]$ reg. chains. Assume we have *solved* F as $V(F) = W(T^i) \cup \cdots \cup W(T^e)$.
- Assume that we have an operation $(f, T) \longmapsto \mathbf{Intersect}(f, T) = (C_1, \ldots, C_d)$ such that

$$V(f) \cap W(T) \ \subseteq \ \cup_i W(C_i) \ \subseteq \ V(f) \cap \overline{W(T)}.$$

Then solving $F \cup f$ reduces to $\mathbf{Intersect}(f, T^i)$ for all i.

Incremental Solving

- Let $F \subset \mathbf{k}[\mathbf{x}]$, $f \in \mathbf{k}[\mathbf{x}]$, $T, T^m \ldots, T^e \subset \mathbf{k}[\mathbf{x}]$ reg. chains. Assume we have *solved* F as $V(F) = W(T^i) \cup \cdots \cup W(T^e)$.
- Assume that we have an operation $(f, T) \longmapsto \textbf{Intersect}(f, T) = (C_1, \ldots, C_d)$ such that

$$V(f) \cap W(T) \;\subseteq\; \cup_i W(C_i) \;\subseteq\; V(f) \cap \overline{W(T)}.$$

Then solving $F \cup f$ reduces to $\textbf{Intersect}(f, T^i)$ for all i.

$\Longrightarrow$ the core routine operates on well behaved objects.

Incremental Solving

- Let $F \subset \mathbf{k}[\mathbf{x}]$, $f \in \mathbf{k}[\mathbf{x}]$, $T, T^m \ldots, T^e \subset \mathbf{k}[\mathbf{x}]$ reg. chains. Assume we have *solved* F as $V(F) = W(T^i) \cup \cdots \cup W(T^e)$.
- Assume that we have an operation $(f, T) \longmapsto \mathbf{Intersect}(f, T) = (C_1, \ldots, C_d)$ such that

$$V(f) \cap W(T) \ \subseteq \ \cup_i W(C_i) \ \subseteq \ V(f) \cap \overline{W(T)}.$$

Then solving $F \cup f$ reduces to $\mathbf{Intersect}(f, T^i)$ for all i.

$\Longrightarrow$ the core routine operates on well behaved objects.

$\Longrightarrow$ the decomposition can be reduced to regular GCD computation, allowing modular methods and fast arithmetic.

Incremental Solving

- Let $F \subset \mathbf{k}[\mathbf{x}]$, $f \in \mathbf{k}[\mathbf{x}]$, $T, T^m \ldots, T^e \subset \mathbf{k}[\mathbf{x}]$ reg. chains. Assume we have *solved* F as $V(F) = W(T^i) \cup \cdots \cup W(T^e)$.
- Assume that we have an operation $(f, T) \longmapsto \mathbf{Intersect}(f, T) = (C_1, \ldots, C_d)$ such that

$$V(f) \cap W(T) \;\subseteq\; \cup_i W(C_i) \;\subseteq\; V(f) \cap \overline{W(T)}.$$

Then solving $F \cup f$ reduces to $\mathbf{Intersect}(f, T^i)$ for all i.

$\Longrightarrow$ the core routine operates on well behaved objects.

$\Longrightarrow$ the decomposition can be reduced to regular GCD computation, allowing modular methods and fast arithmetic.

Related Work

(D. Lazard 91) proposes the principle. (M. M. M. 00) introduces regular GCDs and gives a complete incremental algorithm which, in addition, generates components by decreasing order of dimension.

The notion of a Regular GCD

- Let $P, Q, G \in \mathbf{k}[x_1 < \cdots < x_n][y]$ and $T \subset \mathbf{k}[x_1 < \cdots < x_n]$ reg. chain. G is a *regular GCD* of P, Q modulo $\mathrm{sat}(T)$ if
  - *(i)* $\mathrm{lc}(G, y)$ is a regular modulo $\mathrm{sat}(T)$,
  - *(ii)* $G \in \langle P, Q \rangle$ modulo $\mathrm{sat}(T)$,
  - *(iii)* $\deg_y(G) > 0 \;\Rightarrow\; \mathrm{prem}_y(P, G), \mathrm{prem}_y(Q, G) \in \mathrm{sat}(T)$.

The notion of a Regular GCD

- Let $P, Q, G \in \mathbf{k}[x_1 < \cdots < x_n][y]$ and $T \subset \mathbf{k}[x_1 < \cdots < x_n]$ reg. chain. G is a *regular GCD* of P, Q modulo $\mathrm{sat}(T)$ if
  - (*i*) $\mathrm{lc}(G, y)$ is a regular modulo $\mathrm{sat}(T)$,
  - (*ii*) $G \in \langle P, Q \rangle$ modulo $\mathrm{sat}(T)$,
  - (*iii*) $\deg_y(G) > 0 \;\Rightarrow\; \mathrm{prem}_y(P, G), \mathrm{prem}_y(Q, G) \in \mathrm{sat}(T)$.
- If both $T \cup P$ and $T \cup Q$ are regular chains and if G is a GCD of P, Q modulo $\mathrm{sat}(T)$ with $\deg_y(G) > 0$ then we have

$$\begin{array}{rl} W(T \cup P) \cap V(Q) \subseteq & W(T \cup G) \cup \\ & W(T \cup P) \cap V(Q, h_G) \subseteq \overline{W(T \cup P)} \cap V(Q). \end{array}$$

The notion of a Regular GCD

- Let $P, Q, G \in \mathbf{k}[x_1 < \cdots < x_n][y]$ and $T \subset \mathbf{k}[x_1 < \cdots < x_n]$ reg. chain. G is a *regular GCD* of P, Q modulo $\mathrm{sat}(T)$ if
  - (i) $\mathrm{lc}(G, y)$ is a regular modulo $\mathrm{sat}(T)$,
  - (ii) $G \in \langle P, Q \rangle$ modulo $\mathrm{sat}(T)$,
  - (iii) $\deg_y(G) > 0 \ \Rightarrow\ \mathrm{prem}_y(P, G), \mathrm{prem}_y(Q, G) \in \mathrm{sat}(T)$.
- If both $T \cup P$ and $T \cup Q$ are regular chains and if G is a GCD of P, Q modulo $\mathrm{sat}(T)$ with $\deg_y(G) > 0$ then we have

$$\begin{aligned} W(T \cup P) \cap V(Q) \subseteq\ & W(T \cup G) \cup \\ & W(T \cup P) \cap V(Q, h_G) \subseteq \overline{W(T \cup P)} \cap V(Q). \end{aligned}$$

- One can compute $T^1, \ldots, T^e$ and $G_1, \ldots, G_e$ such that G_i is a reg. GCD of P, Q mod $\mathrm{sat}(T_i)$ and $\sqrt{\mathrm{sat}(T)} = \cap_{i=0}^{e} \sqrt{\mathrm{sat}(T^i)}$.

The notion of a Regular GCD

- Let $P, Q, G \in \mathbf{k}[x_1 < \cdots < x_n][y]$ and $T \subset \mathbf{k}[x_1 < \cdots < x_n]$ reg. chain. G is a *regular GCD* of P, Q modulo $\mathrm{sat}(T)$ if
  - (i) $\mathrm{lc}(G, y)$ is a regular modulo $\mathrm{sat}(T)$,
  - (ii) $G \in \langle P, Q \rangle$ modulo $\mathrm{sat}(T)$,
  - (iii) $\deg_y(G) > 0 \;\Rightarrow\; \mathrm{prem}_y(P, G), \mathrm{prem}_y(Q, G) \in \mathrm{sat}(T)$.
- If both $T \cup P$ and $T \cup Q$ are regular chains and if G is a GCD of P, Q modulo $\mathrm{sat}(T)$ with $\deg_y(G) > 0$ then we have

$$
\begin{aligned}
W(T \cup P) \cap V(Q) \subseteq \;& W(T \cup G) \cup \\
& W(T \cup P) \cap V(Q, h_G) \subseteq \overline{W(T \cup P)} \cap V(Q).
\end{aligned}
$$

- One can compute $T^1, \ldots, T^e$ and $G_1, \ldots, G_e$ such that G_i is a reg. GCD of P, Q mod $\mathrm{sat}(T_i)$ and $\sqrt{\mathrm{sat}(T)} = \cap_{i=0}^{e} \sqrt{\mathrm{sat}(T^i)}$.

Related Work

(M. Kalkbrener 91) (M. M. M., Renaud Rioboo 95) (M. M. M. 00)

Regular GCDs: Bottom-up or Top-down?

- Let $P, Q \in \mathbf{k}[x_1 < \cdots < x_n][y]$ and $T \subset \mathbf{k}[x_1 < \cdots < x_n]$ reg. chain. How to compute a *regular GCD* of P, Q mod $\mathrm{sat}(T)$?
- (M. Kalkbrener 91) uses Pseudo-Remainder Sequences. Inefficient!

Regular GCDs: Bottom-up or Top-down?

- Let $P, Q \in \mathbf{k}[x_1 < \cdots < x_n][y]$ and $T \subset \mathbf{k}[x_1 < \cdots < x_n]$ reg. chain. How to compute a *regular GCD* of P, Q mod $\mathrm{sat}(T)$?
- (M. Kalkbrener 91) uses Pseudo-Remainder Sequences. Inefficient!
- (Jean Della Dora, Claire Dicrescenzo, Dominique Duval 85) assume $\mathrm{sat}(T)$ radical and compute (naively) the subresultant chain of P, Q in $\mathbf{k}[x_1 < \cdots < x_n][y]$. Limited and practically inefficient!

Regular GCDs: Bottom-up or Top-down?

- Let $P, Q \in \mathbf{k}[x_1 < \cdots < x_n][y]$ and $T \subset \mathbf{k}[x_1 < \cdots < x_n]$ reg. chain. How to compute a *regular GCD* of P, Q mod $\mathrm{sat}(T)$?
- (M. Kalkbrener 91) uses Pseudo-Remainder Sequences. Inefficient!
- (Jean Della Dora, Claire Dicrescenzo, Dominique Duval 85) assume $\mathrm{sat}(T)$ radical and compute (naively) the subresultant chain of P, Q in $\mathbf{k}[x_1 < \cdots < x_n][y]$. Limited and practically inefficient!
- (M. M. M. and R. Rioboo 95) assume $\mathrm{sat}(T)$ radical $+$ 0-dimensional and use the subresultant chain of P, Q directly in $\mathbf{k}[x_1 < \cdots < x_n][y]$ mod $\mathrm{sat}(T)$. Better but removing the assumptions removes the efficiency.

Equiprojectable Decomposition (1/2)

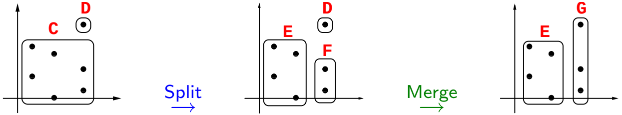
$$C \left| \begin{array}{l} C_2 = y^2 + 6yx^2 + 2y + x \\ C_1 = x^3 + 6x^2 + 5x + 2 \end{array} \right. , \quad D \left| \begin{array}{l} D_2 = y + 6 \\ D_1 = x + 6 \end{array} \right.$$

$\downarrow$ Split C : GCD $\downarrow$

$$E \left| \begin{array}{l} {C_2}' = y^2 + x \\ {C_1}' = x^2 + 5 \end{array} \right. , \quad F \left| \begin{array}{l} C_2'' = y^2 + y + 1 \\ C_1'' = x + 6 \end{array} \right. , \quad D \left| \begin{array}{l} D_2 = y + 6 \\ D_1 = x + 6 \end{array} \right.$$

$\downarrow$ Merge F and D : CRT $\downarrow$

$$E \left| \begin{array}{l} C_2' = y^2 + x \\ C_1' = x^2 + 5 \end{array} \right. , \quad G \left| \begin{array}{l} G_2 = y^3 + 6 \\ G_1 = x + 6 \end{array} \right.$$



Equiprojectable Decomposition (2/2)

$$
\begin{tikzcd}
\begin{array}{c} F \subset \mathbb{Q}[x_1, \ \ldots, \ x_n] \\ p \text{ prime, } p > A(F) \end{array} \arrow[r] \arrow[d, "\text{Triang.}(F \mod p)"'] & Eq.D._{\mathbb{Q}}(F) \\
\begin{array}{c} \{T^1, \ldots, \ T^s\} \\ \subset \mathbb{F}_p[x_1, \ \ldots, \ x_n] \end{array} \arrow[r, "\textbf{Split + Merge}"] & Eq.D._{\mathbb{F}_p}(F) \arrow[u, "\textit{Lifting}"']
\end{tikzcd}
$$

- $A(F) := 2n^2 d^{2n+1}(3h + 7\log(n+1) + 5n \log d + 10)$ where h and d upper bound coeff. sizes and total degrees for $f \in F$. Assumes F square and generates a 0-dimensional radical ideal.

Equiprojectable Decomposition (2/2)

$$
\begin{tikzcd}
\begin{array}{c} F \subset \mathbb{Q}[x_1, \ldots, x_n] \\ p \text{ prime}, \ {\color{blue} p > A(F)} \end{array} \arrow[r] \arrow[d, "\text{Triang.}(F \mod p)"'] & Eq.D._{\mathbb{Q}}(F) \\
\begin{array}{c} \{T^1, \ldots, T^s\} \\ \subset \mathbb{F}_p[x_1, \ldots, x_n] \end{array} \arrow[r, "{\color{blue}\textbf{Split + Merge}}"] & Eq.D._{\mathbb{F}_p}(F) \arrow[u, "{\color{blue}\textit{Lifting}}"']
\end{tikzcd}
$$

- $A(F) := 2n^2 d^{2n+1}(3h + 7\log(n+1) + 5n\log d + 10)$ where h and d upper bound coeff. sizes and total degrees for $f \in F$. Assumes F square and generates a 0-dimensional radical ideal.
- If $p \nmid A(F)$, the equiprojectable decomposition specializes well mod p.

Equiprojectable Decomposition (2/2)

$$
\begin{tikzcd}
\begin{array}{c} F \subset \mathbb{Q}[x_1, \ldots, x_n] \\ p \text{ prime}, \ p > A(F) \end{array} \arrow[rr] \arrow[d, "\text{Triang.}(F \mod p)"'] & & Eq.D._{\mathbb{Q}}(F) \\
\begin{array}{c} \{T^1, \ldots, T^s\} \\ \subset \mathbb{F}_p[x_1, \ldots, x_n] \end{array} \arrow[rr, "\textbf{Split + Merge}"] & & Eq.D._{\mathbb{F}_p}(F) \arrow[u, "\textit{Lifting}"']
\end{tikzcd}
$$

- $A(F) := 2n^2 d^{2n+1}(3h + 7\log(n+1) + 5n \log d + 10)$ where h and d upper bound coeff. sizes and total degrees for $f \in F$. Assumes F square and generates a 0-dimensional radical ideal.
- If $p \not| A(F)$, the equiprojectable decomposition specializes well mod p.
- In practice we choose p much smaller with a probability of success, i.e. $> 99\%$ with $p \approx ln(A(F))$ (Xavier Dahan, M. M. M., Éric Schost, Wenyuan Wu, Yuzhen Xie 05).

Fast Polynomial Arithmetic (1/2)

- Let A and B in $\mathbf{k}[x_1, \ldots, x_n]$ reduced w.r.t. $T := \{T_1, \ldots, T_n\}$ 0-dimensional reg. chain with all $\mathrm{init}(T_i) = 1$.
- The size of input is $\delta_{\mathbf{T}} = \deg(T_1, x_1) \cdots \deg(T_n, x_n)$.

Fast Polynomial Arithmetic (1/2)

- Let A and B in $\mathbf{k}[x_1, \ldots, x_n]$ reduced w.r.t. $T := \{T_1, \ldots, T_n\}$ 0-dimensional reg. chain with all $\mathrm{init}(T_i) = 1$.
- The size of input is $\delta_{\mathbf{T}} = \deg(T_1, x_1) \cdots \deg(T_n, x_n)$.
- One can compute $A\,B \mod \langle T_1, \ldots, T_n\}$ in $O\tilde{}(4^n \delta_{\mathbf{T}})$ operations in $\mathbf{k}$ (Xin Li, M.M.M., É. Schost 07).

Fast Polynomial Arithmetic (1/2)

- Let A and B in $\mathbf{k}[x_1, \ldots, x_n]$ reduced w.r.t. $T := \{T_1, \ldots, T_n\}$ 0-dimensional reg. chain with all $\mathrm{init}(T_i) = 1$.
- The size of input is $\delta_{\mathbf{T}} = \deg(T_1, x_1) \cdots \deg(T_n, x_n)$.
- One can compute $A\,B \mod \langle T_1, \ldots, T_n\}$ in $O\tilde{}(4^n \delta_{\mathbf{T}})$ operations in $\mathbf{k}$ (Xin Li, M.M.M., É. Schost 07).
- Two key ideas: using the fast division trick of (Sieveking 72) (Kung 74) and avoid $\mod \langle T_1, \ldots, T_n\}$ as much as possible.

$\underline{\mathrm{ModMul}(A, B, \{T_1, \ldots, T_n\})}$

1 $D := AB$ computed in $\mathbf{k}[x_1, \ldots, x_n]$

2 **return** $\mathrm{NormalForm}_n(D, \{T_1, \ldots, T_n\})$

Fast Polynomial Arithmetic (2/2)

$\mathsf{NormalForm}_1(A : R[x_1], \{T_1 : R[x_1]\})$

$S_1 := \mathrm{Rev}(T_1)^{-1} \mod x_1^{\deg(A)-\deg(T_1)+1}$

$D := \mathrm{Rev}(A) S_1 \mod x_1^{\deg(A)-\deg(T_1)+1}$

$D := T_1 \, \mathrm{Rev}(D)$

return $A - D$

$\mathsf{NormalForm}_2(A : R[x_1, x_2], \{T_1 : R[x_1], T_2 : R[x_1, x_2]\})$

$A := \mathrm{map}(\mathsf{NormalForm}_1, \mathrm{Coeffs}(A, x_2), \{T_1\})$

$S_2 := \mathrm{Rev}(T_2)^{-1} \mod T_1, x_2^{\deg(A,x_2)-\deg(T_2,x_2)+1}$

$D := \mathrm{Rev}(A) S_2 \mod x_2^{\deg(A,x_2)-\deg(T_2,x_2)+1}$

$D := \mathrm{map}(\mathsf{NormalForm}_1, \mathrm{Coeffs}(D, x_2), \{T_1\})$

$D := T_2 \, \mathrm{Rev}(D)$

$D := \mathrm{map}(\mathsf{NormalForm}_1, \mathrm{Coeffs}(D, x_2), \{T_1\})$

return $A - D$

Subresultants and Regular GCDs

- Let $P, Q \in \mathbf{k}[x_1 < \cdots < x_n][y]$ with main variable y.
- Let S_j for the j-th subresultant (w.r.t. y) of P, Q. Let $T \subset \mathbf{k}[x_1 < \cdots < x_n]$ be regular chain.

Subresultants and Regular GCDs

- Let $P, Q \in \mathbf{k}[x_1 < \cdots < x_n][y]$ with main variable y.
- Let S_j for the j-th subresultant (w.r.t. y) of P, Q. Let $T \subset \mathbf{k}[x_1 < \cdots < x_n]$ be regular chain.
- Assume
  - $\mathrm{res}(P, Q, y) \in \mathrm{sat}(T)$,
  - $\mathrm{init}(P)$ and $\mathrm{init}(Q)$ are regular modulo $\mathrm{sat}(T)$,
  - Let $1 \leq d \leq \deg(Q, y)$ such that $S_j \in \mathrm{sat}(T)$ for all $0 \leq j < d$.
  - $\mathrm{lc}(S_d, y)$ is regular modulo $\mathrm{sat}(T)$,

Subresultants and Regular GCDs

- Let $P, Q \in \mathbf{k}[x_1 < \cdots < x_n][y]$ with main variable y.
- Let S_j for the j-th subresultant (w.r.t. y) of P, Q. Let $T \subset \mathbf{k}[x_1 < \cdots < x_n]$ be regular chain.
- Assume
  - $\mathrm{res}(P, Q, y) \in \mathrm{sat}(T)$,
  - $\mathrm{init}(P)$ and $\mathrm{init}(Q)$ are regular modulo $\mathrm{sat}(T)$,
  - Let $1 \le d \le \deg(Q, y)$ such that $S_j \in \mathrm{sat}(T)$ for all $0 \le j < d$.
  - $\mathrm{lc}(S_d, y)$ is regular modulo $\mathrm{sat}(T)$,

Theorem (Xin Li, M.M.M., Wei Pan 2009)

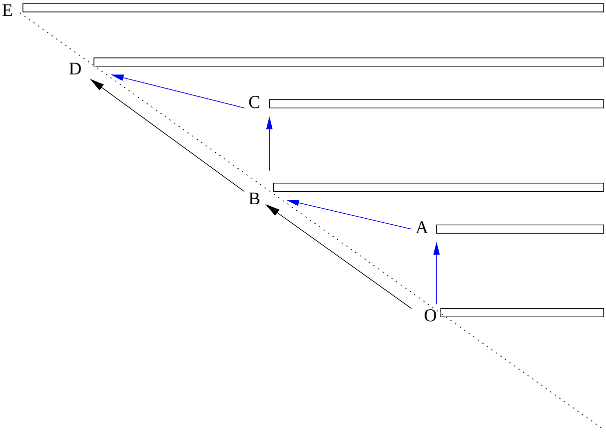
Assume that one of the following conditions holds:

- $\mathrm{sat}(T)$ *is radical,*
- *for all* $d < k \le \mathrm{mdeg}(Q)$, *the coefficient of* y^k *in* S_k *is either null or regular modulo* $\mathrm{sat}(T)$.

Then, S_d *is a regular GCD of* P, Q *modulo* $\mathrm{sat}(T)$.

Computing Regular GCDs: Bottom-up!

- Assume that the subresultants S_j for $1 \leq j < \mathrm{mdeg}(Q)$ are computed.
- Then one can compute a regular GCD of P, Q modulo $\mathrm{sat}(T)$ by performing a bottom-up search.



Computing Regular GCDs: Complexity Estimates!

- Let $x_{n+1} := y$. Define $d_i := \max(\deg(P, x_i), \deg(Q, x_i))$. Define $b_i := 2 d_i d_{n+1}$ and $B := (b_1 + 1) \cdots (b_n + 1)$.

Computing Regular GCDs: Complexity Estimates!

- Let $x_{n+1} := y$. Define $d_i := \max(\deg(P, x_i), \deg(Q, x_i))$. Define $b_i := 2 d_i d_{n+1}$ and $B := (b_1 + 1) \cdots (b_n + 1)$.
- We compute S_j for $1 \leq j < \mathrm{mdeg}(Q)$ via FFT on an n-dim. grid of points not cancelling $\mathrm{init}(P)$ and $\mathrm{init}(Q)$ in

$$O(d_{n+1} B \log(B) + d_{n+1}^2 B) \quad \text{where} \quad B \in O(2^n d_{n+1}^n d_1 \ldots d_n).$$

Computing Regular GCDs: Complexity Estimates!

- Let $x_{n+1} := y$. Define $d_i := \max(\deg(P, x_i), \deg(Q, x_i))$. Define $b_i := 2 d_i d_{n+1}$ and $B := (b_1 + 1) \cdots (b_n + 1)$.
- We compute S_j for $1 \leq j < \mathrm{mdeg}(Q)$ via FFT on an n-dim. grid of points not cancelling $\mathrm{init}(P)$ and $\mathrm{init}(Q)$ in

$$O(d_{n+1} B \log(B) + d_{n+1}^2 B) \quad \text{where} \quad B \in O(2^n d_{n+1}^n d_1 \ldots d_n).$$

- Then $\mathrm{res}(P, Q, y) = S_0$ is interpolated in time $O(B \log(B))$.

Computing Regular GCDs: Complexity Estimates!

- Let $x_{n+1} := y$. Define $d_i := \max(\deg(P, x_i), \deg(Q, x_i))$. Define $b_i := 2 d_i d_{n+1}$ and $B := (b_1 + 1) \cdots (b_n + 1)$.
- We compute S_j for $1 \leq j < \mathrm{mdeg}(Q)$ via FFT on an n-dim. grid of points not cancelling $\mathrm{init}(P)$ and $\mathrm{init}(Q)$ in

$$O(d_{n+1} B \log(B) + d_{n+1}^2 B) \quad \text{where} \quad B \in O(2^n d_{n+1}^n d_1 \ldots d_n).$$

- Then $\mathrm{res}(P, Q, y) = S_0$ is interpolated in time $O(B \log(B))$.
- If $\mathrm{sat}(T)$ is radical, a regular GCD is interpolated within $O(d_{n+1} B \log(B))$ otherwise $O(d_{n+1}^2 B \log(B))$.

Computing Regular GCDs: Complexity Estimates!

- Let $x_{n+1} := y$. Define $d_i := \max(\deg(P, x_i), \deg(Q, x_i))$. Define $b_i := 2 d_i d_{n+1}$ and $B := (b_1 + 1) \cdots (b_n + 1)$.
- We compute S_j for $1 \leq j < \mathrm{mdeg}(Q)$ via FFT on an n-dim. grid of points not cancelling $\mathrm{init}(P)$ and $\mathrm{init}(Q)$ in

$$O(d_{n+1} B \log(B) + d_{n+1}^2 B) \quad \text{where} \quad B \in O(2^n d_{n+1}^n d_1 \ldots d_n).$$

- Then $\mathrm{res}(P, Q, y) = S_0$ is interpolated in time $O(B \log(B))$.
- If $\mathrm{sat}(T)$ is radical, a regular GCD is interpolated within $O(d_{n+1} B \log(B))$ otherwise $O(d_{n+1}^2 B \log(B))$.
- Regularity tests (and normal forms) also fit these bounds.

Computing Regular GCDs: Complexity Estimates!

- Let $x_{n+1} := y$. Define $d_i := \max(\deg(P, x_i), \deg(Q, x_i))$. Define $b_i := 2 d_i d_{n+1}$ and $B := (b_1 + 1) \cdots (b_n + 1)$.
- We compute S_j for $1 \leq j < \mathrm{mdeg}(Q)$ via FFT on an n-dim. grid of points not cancelling $\mathrm{init}(P)$ and $\mathrm{init}(Q)$ in

$$O(d_{n+1} B \log(B) + d_{n+1}^2 B) \quad \text{where} \quad B \in O(2^n d_{n+1}^n d_1 \ldots d_n).$$

- Then $\mathrm{res}(P, Q, y) = S_0$ is interpolated in time $O(B \log(B))$.
- If $\mathrm{sat}(T)$ is radical, a regular GCD is interpolated within $O(d_{n+1} B \log(B))$ otherwise $O(d_{n+1}^2 B \log(B))$.
- Regularity tests (and normal forms) also fit these bounds.
- Best known results for practical sizes, say $d_{n+1} < 500$.

Computing Regular GCDs: Complexity Estimates!

- Let $x_{n+1} := y$. Define $d_i := \max(\deg(P, x_i), \deg(Q, x_i))$. Define $b_i := 2 d_i d_{n+1}$ and $B := (b_1 + 1) \cdots (b_n + 1)$.
- We compute S_j for $1 \leq j < \mathrm{mdeg}(Q)$ via FFT on an n-dim. grid of points not cancelling $\mathrm{init}(P)$ and $\mathrm{init}(Q)$ in

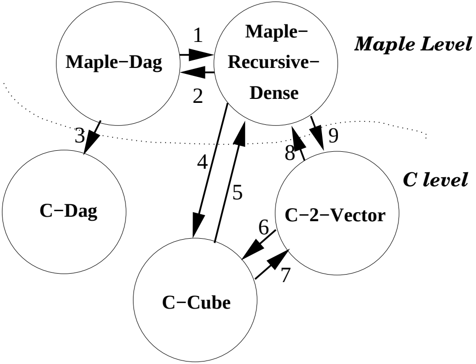
$$O(d_{n+1} B \log(B) + d_{n+1}^2 B) \quad \text{where} \quad B \in O(2^n d_{n+1}^n d_1 \ldots d_n).$$

- Then $\mathrm{res}(P, Q, y) = S_0$ is interpolated in time $O(B \log(B))$.
- If $\mathrm{sat}(T)$ is radical, a regular GCD is interpolated within $O(d_{n+1} B \log(B))$ otherwise $O(d_{n+1}^2 B \log(B))$.
- Regularity tests (and normal forms) also fit these bounds.
- Best known results for practical sizes, say $d_{n+1} < 500$.
- If a regular GCD is expected to have degree 1 in y all computations fit in $O\tilde{}(d_{n+1} B)$ which is essentially optimal.

The RegularChains library in MAPLE

- 80,000 lignes of MAPLE code, 36,000 lignes of C code, 121 Commands, 6 modules ChainTools, MatrixTools, ConstructibleSetTools. ParametricSystemTools, SemiAlgebraicSetTools , FastArithmeticTools .
- Main new commnands in MAPLE 13: IsPrimitive, ComplexRootClassification, RealRootClassification, RealRootIsolate RealRootCounting, BorderPolynomial, + those of FastArithmeticTools (see demo).
- Current contributors: Changbo Chen, Francçois Lemaire, Liyun Li, Xin Li, M.M.M., Wei Pan, Bican Xia, Rong Xiao, Yuzhen Xie.

The MODPN library

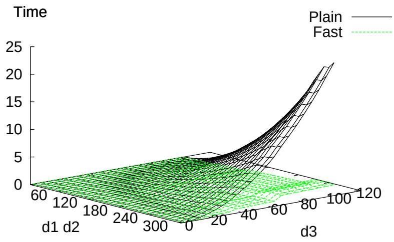


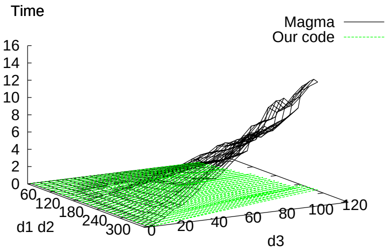
- *C-Dag* for straight-line program.
- *C-Cube* for FFT-based computations.
- *C-2-Vector* for compact dense representation.
- *Maple-Dag* for calling `RegularChains` library.
- *Maple-Recursive-Dense* for calling RECDEN library.

Fast Normal Form Benchmarks

[left] comparison of classical (plain) and asymptotically fast strategies.

[right] comparison with MAGMA.

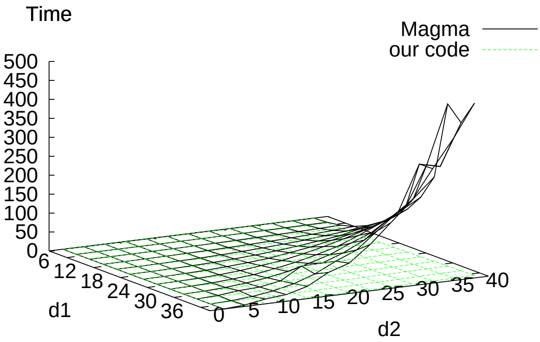




- Asymptotically fast strategy dominates the classical one.
- Our fast implementation is better than Magma's one (the best known implementation).

Generic Bivariate Systems

- "our code" means `BivariateModularTriangularize` in MAPLE 13.
- Random generic input systems, thus equiprojectable.
- For the largest examples (having about 5700 solutions), the ratio is about 460/7 in our favor.



Non-generic Bivariate Systems

- Examples designed to enforce many "splittings" (many equiprojectable components).
- For the largest examples, the ratio is 5260/80, in our favor.

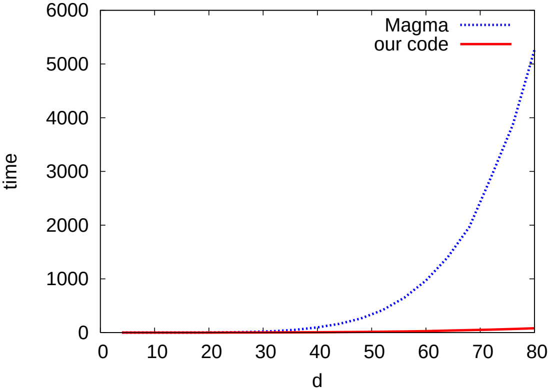


Figure: Non-generic bivariate systems: MAGMA vs. us.

Generic Trivariate Systems

- MAPLE means the experimental and fast version of `Triangularize` to be integrated in MAPLE 14.

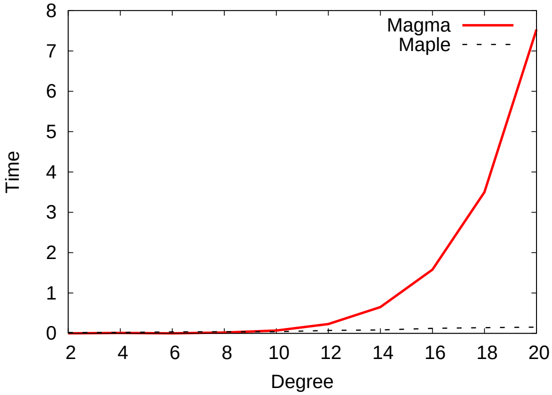


Figure: *Generic dense 3-variable.*

Regularity Test ($=$ Saturation)

d_1	d_2	d_3	Regularize	Fast Regularize	Magma
2	2	3	0.032	0.004	0.010
3	4	6	0.160	0.016	0.020
4	6	9	0.404	0.024	0.060
5	8	12	$>$100	0.129	0.330
6	10	15	$>$100	0.272	1.300
7	12	18	$>$100	0.704	5.100
8	14	21	$>$100	1.276	14.530
9	16	24	$>$100	5.836	40.770
10	18	27	$>$100	9.332	107.280
11	20	30	$>$100	15.904	229.950
12	22	33	$>$100	33.146	493.490

Table: Generic dense 3-variable.

- In the non-generic case, both gaps are even larger.
- "Fast Regularize" means `RegularizeDim0` in MAPLE 13.

Do Triangular Decompositions Have the Rigth Size?

Do Triangular Decompositions Have the Rigth Size?

- Let $T \subset \mathbf{k}[x_1, \ldots, x_n]$ be a 0-dimensional reg. chain with all $\mathrm{init}(T_i) = 1$. There exists a 0-dimensional reg. chain N such that $V(T) = V(N)$ and the height (or "size") of each coefficient in N is upper bounded by
  - the height of $\mathbf{V}(T)$, that is, the minimum size of a data set encoding $\mathbf{V}(T)$, if $\mathbf{k} = \mathbb{Q}$,
  - the degree of $\mathbf{V}(T^{\downarrow})$, if $\mathbf{k}$ is a field $\mathbb{F}_p(t_1, \ldots, t_m)$ of rational functions and $T^{\downarrow}$ is T regarded in $\mathbf{k}[t_1, \ldots, t_m, x_1, \ldots, x_n]$.

  See (X. Dahan, É. Schost 04) for precise statements.

Do Triangular Decompositions Have the Rigth Size?

- Let $T \subset \mathbf{k}[x_1, \ldots, x_n]$ be a 0-dimensional reg. chain with all $\mathrm{init}(T_i) = 1$. There exists a 0-dimensional reg. chain N such that $V(T) = V(N)$ and the height (or "size") of each coefficient in N is upper bounded by
  - the height of $\mathbf{V}(T)$, that is, the minimum size of a data set encoding $\mathbf{V}(T)$, if $\mathbf{k} = \mathbb{Q}$,
  - the degree of $\mathbf{V}(T^{\downarrow})$, if $\mathbf{k}$ is a field $\mathbb{F}_p(t_1, \ldots, t_m)$ of rational functions and $T^{\downarrow}$ is T regarded in $\mathbf{k}[t_1, \ldots, t_m, x_1, \ldots, x_n]$.

  See (X. Dahan, É. Schost 04) for precise statements.

- Let $\mathcal{I} \subset \mathbf{k}[x_1 < x_2]$ be 0-dimensional radical with degree d and h be the height of $V(\mathcal{I})$. There exists a (non-reduced) lexicographical Gröbner basis G of $\mathcal{I}$ such that the height of a coefficient is essentially quadratic in both h and d. These estimatas are sharp. Estimates become cubic for the reduced basis. (X. Dahan 09).

What Plays the Role of Degree Bases for Triangular Decompositions ?

What Plays the Role of Degree Bases for Triangular Decompositions ?

- In my opinionm Triangular Decompositions themselves do:
  - Even if degree bases instead of lex bases are used in the previous benchmarks, the bivariate and triavariate `Triangularize` solvers remain faster.
  - Keep in mind that a triangular decomposition is "essentially" a factored lex basis (D. Lazard 92)
  - For Generic input system $\mathbb{F}_p[x_1, \ldots, x_n]$ in degree $d = 2$,

$n = 3$ the output of `Triangularize` vs that of `Basis` gies 7200 vs 13400 characters long

$n = 4$ the output of `Triangularize` vs that of `Basis` gies 23000 vs 1005000.

What Plays the Role of Degree Bases for Triangular Decompositions ?

- In my opinionm Triangular Decompositions themselves do:
  - Even if degree bases instead of lex bases are used in the previous benchmarks, the bivariate and triavariate `Triangularize` solvers remain faster.
  - Keep in mind that a triangular decomposition is "essentially" a factored lex basis (D. Lazard 92)
  - For Generic input system $\mathbb{F}_p[x_1, \ldots, x_n]$ in degree $d = 2$,

    $n = 3$ the output of `Triangularize` vs that of `Basis` gies 7200 vs 13400 characters long

    $n = 4$ the output of `Triangularize` vs that of `Basis` gies 23000 vs 1005000.

- Size estimates are promising but not enough. More on this another time.

Can Triangular Decomposition Preserve Multiplicities?

Can Triangular Decomposition Preserve Multiplicities?

- See previous talk.

Which Types of Polynomial Systems for Which Method?

Which Types of Polynomial Systems for Which Method?

- Incremental Methods are probably best for square systems, regular sequences and probably bad for overconstrained systems.

Which Types of Polynomial Systems for Which Method?

- Incremental Methods are probably best for square systems, regular sequences and probably bad for overconstrained systems.

- For overconstrained systems one may conider restarting from the elimnation methods (Wen-Tsün Wu, Dongming Wang, Kalkbrener, ...) making use of modular methods, fast arithmetic, multivariate resultants, etc.

Can we unify terminology?

Xie Xie! Thank You!