# Improved method for finding optimal formulae for bilinear maps

Svyatoslav Covanov

Team CARAMBA

July 20, 2017




Optimal formulae

Exhaustive search algorithms

Technical aspects and experimental results

# Short product example

How to multiply two polynomials $A = a_0 + a_1 X + a_2 X^2$ and $B = b_0 + b_1 X + b_2 X^2$ modulo $X^3$ ?

# Short product example

How to multiply two polynomials $A = a_0 + a_1X + a_2X^2$ and $B = b_0 + b_1X + b_2X^2$ modulo $X^3$ ?

$A \cdot B = a_0b_0 + (a_1b_0 + a_0b_1)X + (a_2b_0 + a_1b_1 + a_0b_2)X^2$

1. **Naive multiplication:**
   - $\pi_0 = \boldsymbol{a_0b_0}$, $\pi_1 = \boldsymbol{a_1b_0}$, $\pi_2 = \boldsymbol{a_0b_1}$, $\pi_3 = \boldsymbol{a_1b_1}$, $\pi_4 = \boldsymbol{a_2b_1}$ and $\pi_5 = \boldsymbol{a_1b_2}$.
   - We have $A \cdot B = \pi_0 + (\pi_1 + \pi_2)X + (\pi_3 + \pi_4 + \pi_5)X^2$.

2. **Optimal formula**:
   - $\pi_0 = \boldsymbol{a_0b_0}$, $\pi_1 = \boldsymbol{a_1b_1}$, $\pi_2 = \boldsymbol{a_2b_2}$, $\pi_3 = \boldsymbol{(a_0 + a_1)(b_0 + b_1)}$ and $\pi_4 = \boldsymbol{(a_0 + a_2)(b_0 + b_2)}$.
   - We have $A \cdot B = \pi_0 + (\pi_3 - \pi_0 - \pi_1)X + (\pi_1 + \pi_4 - \pi_0 - \pi_2)X^2$.

   The **bilinear rank** is equal to 5.

# Matrix formalism

$$c_0 = \begin{pmatrix} a_0 & a_1 & a_2 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = a_0 b_0$$

$$c_1 = \begin{pmatrix} a_0 & a_1 & a_2 \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = a_1 b_0 + a_0 b_1$$

$$c_2 = \begin{pmatrix} a_0 & a_1 & a_2 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} b_0 \\ b_1 \\ b_2 \end{pmatrix} = a_2 b_0 + a_1 b_1 + a_0 b_2$$

## Matrix representation of formulae:

$$\underbrace{\begin{pmatrix}1&0&0\\0&0&0\\0&0&0\end{pmatrix}}_{a_0b_0},\ \underbrace{\begin{pmatrix}0&0&0\\0&1&0\\0&0&0\end{pmatrix}}_{a_1b_1}\ \underbrace{\begin{pmatrix}0&0&0\\0&0&0\\0&0&1\end{pmatrix}}_{a_2b_2}\ \underbrace{\begin{pmatrix}1&1&0\\1&1&0\\0&0&0\end{pmatrix}}_{(a_0+a_1)(b_0+b_1)}\ \underbrace{\begin{pmatrix}1&0&1\\0&0&0\\1&0&1\end{pmatrix}}_{(a_0+a_2)(b_0+b_2)}$$

## Decomposition with rank-one matrices:

$$\underbrace{\begin{pmatrix}0&1&0\\1&0&0\\0&0&0\end{pmatrix}}_{c_1} = \begin{pmatrix}1&1&0\\1&1&0\\0&0&0\end{pmatrix} - \begin{pmatrix}1&0&0\\0&0&0\\0&0&0\end{pmatrix} - \begin{pmatrix}0&0&0\\0&1&0\\0&0&0\end{pmatrix}$$

$$\underbrace{\begin{pmatrix}0&0&1\\0&1&0\\1&0&0\end{pmatrix}}_{c_2} = \begin{pmatrix}0&0&0\\0&1&0\\0&0&0\end{pmatrix} + \begin{pmatrix}1&0&1\\0&0&0\\1&0&1\end{pmatrix} - \begin{pmatrix}1&0&0\\0&0&0\\0&0&0\end{pmatrix} - \begin{pmatrix}0&0&0\\0&0&0\\0&0&1\end{pmatrix}$$

# Problem to be solved

Let $K$ be a finite field. Let $T$ (the target) be a subspace of $\mathcal{M}_{n,n}(K)$ of dimension $\ell$.
Let $\mathcal{G}$ be the set of matrices of rank one in $\mathcal{M}_{n,n}(K)$.
**Problem to be solved:** Find all free families $\mathcal{F} \subset \mathcal{G}$ of minimal size such that $T \subset \mathrm{Span}(\mathcal{F})$.

## Definition

Let $r \geq 0$ and $n \geq 0$. We denote by $\mathscr{S}_r$ the set of all subspaces $V \subset \mathcal{M}_{n,n}$ such that there exists $\{g_0, \ldots, g_{r-1}\}$ a free family of $\mathcal{G}$ satisfying $V = \mathrm{Span}(g_0, \ldots, g_{r-1})$.

**Restatement:**

1. Find minimal $r$ such that $\mathscr{S}_r$ contains subspaces $V$ s.t. $T \subset V$;
2. Enumerate the bases of rank-one matrices for subspaces $V \in \mathscr{S}_r$ s.t. $T \subset V$.

# Short Product Example

For the short product modulo $X^3$, we have

$$T = \mathrm{Span}\left( \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \right),$$

$K = \mathbb{F}_2$, $\ell = n = 3$ and $r = 5$.

# Naive algorithm

Enumerate all subspaces $V \in \mathscr{S}_r$ and keep those which contain T.

Complexity: $\#\mathscr{S}_r \leq \binom{\#\mathcal{G}}{r}$. For $\ell = 3$ and $K = \mathbb{F}_2$, we have

$\#\mathscr{S}_5 = 157,535 \ll 1,906,884 = \binom{49}{5}$.

# Incomplete basis improvement

**Theorem**

*Let $T$ be a subspace of dimension $\ell$ of $\mathcal{M}_{n,n}$, let $r \geq \ell$ be an integer. For any $V \in \mathscr{S}_r$, such that $T \subset V$, there exists $W \in \mathscr{S}_{r-\ell}$ such that $T \oplus W = V$.*

Incomplete basis improvement: compute all the vector spaces $V = T + W$ for $W \in \mathscr{S}_{r-\ell}$ and keep those which are in $\mathscr{S}_r$.

Complexity: $\#\mathscr{S}_{r-\ell} \leq \binom{\#\mathcal{G}}{r-\ell}$. For $\ell = 3$, $\#\mathscr{S}_2 = 980 \ll 157,535$.

# Automorphisms

We consider the action of pairs $(P, Q)$ ($P$ and $Q$ in $\mathrm{GL}_n$) on $M \in \mathcal{M}_{n,n}$:

$$M \circ (P, Q) = P^T \cdot M \cdot Q.$$

Let $\mathrm{Stab}(T)$ be the group of $(P, Q)$ such that

$$\forall M \in T,\ M \circ (P, Q) \in T.$$

The group action can be used with all the previous algorithms: we compute $\mathscr{S}_r / \mathrm{Stab}(T)$ or $\mathscr{S}_{r-\ell} / \mathrm{Stab}(T)$.

# Intermediate strategies

The two previous strategies are two extreme cases of a mixed strategy.

Let $k \geq 0$. For all $W \in \mathscr{S}_{r-\ell+k}$ such that $\dim(W \cap T) = k$, compute the vector spaces $V = T + W$ and keep those which are in $\mathscr{S}_r$.

## Notation

*For an integer $d \geq 0$ and a subspace $F \subset \mathcal{M}_{n,n}$, we denote by $\mathcal{C}_d(F)$ the set of subspaces $W \in \mathscr{S}_d$ such that $F \subset W$.*

# Intermediate strategies

The two previous strategies are two extreme cases of a mixed strategy.

Let $k \geq 0$. For all $W \in \mathscr{S}_{r-\ell+k}$ such that $\dim(W \cap T) = k$, compute the vector spaces $V = T + W$ and keep those which are in $\mathscr{S}_r$.

## Notation

*For an integer $d \geq 0$ and a subspace $F \subset \mathcal{M}_{n,n}$, we denote by $\mathcal{C}_d(F)$ the set of subspaces $W \in \mathscr{S}_d$ such that $F \subset W$.*

- Naive algorithm: compute $\mathcal{C}_r(\emptyset)$;
- Incomplete basis improvement: compute $\mathcal{C}_{r-\ell}(\emptyset)$;
- General case: given $g$ subspaces $F_0, \ldots, F_{g-1}$ of $T$ of dimensions $k_0, \ldots, k_{g-1}$, compute $\mathcal{C}_{r-\ell+k_0}(F_0), \ldots, \mathcal{C}_{r-\ell+k_{g-1}}(F_{g-1})$.

# Example for the short product

## Notation

*For an integer $\ell \geq 0$ , we denote by $T_\ell$ the subspace of $\mathcal{M}_{\ell,\ell}$ such that $T_\ell = \mathsf{Span}(c_0, \ldots, c_{\ell-1})$, where the $c_i$'s are the coefficients of the short product modulo $X^\ell$.*

## Theorem

*Let $V \in \mathscr{S}_r$ containing $T_\ell$. There exist $\sigma \in \mathsf{Stab}(T_\ell)$ and $W \in \mathcal{C}_{r-\ell+2}(\mathsf{Span}(c_{\ell-1}, c_{\ell-2}))$ such that $V = T_\ell + W \circ \sigma$.*

For $\ell = 3$:

| approach | covering set | cardinality |
|---|---|---|
| Naive approach | $\mathcal{C}_5(\emptyset)$ | $157,535$ |
| BDEZ '12 | $\mathcal{C}_2(\emptyset)$ | 980 |
| New approach | $\mathcal{C}_4(\mathsf{Span}(c_2, c_1))$ | 12 |

Optimal formulae

Exhaustive search algorithms

Technical aspects and experimental results

# Statement of the problem

Algorithmic problem: enumerate a set of the form

$$\mathcal{C}_{r-\ell+k}(\mathrm{Span}(\phi_0,\ldots,\phi_{k-1}))/\,\mathrm{Stab}(T)\cap\mathrm{Stab}(\mathrm{Span}(\phi_0,\ldots,\phi_{k-1})),$$

where $T$ is a subspace of $\mathcal{M}_{n,n}$ of dimension $\ell$ and the $\phi_i$'s are elements of $T$.

**Steps:**

- precompute $\mathscr{S}_{r-\ell+k}/\,\mathrm{GL}_n\times GL_n$,
- deduce $\mathcal{C}_{r-\ell+k}(\mathrm{Span}(\phi_0,\ldots,\phi_{k-1}))/\,\mathrm{Stab}(\{\phi_0,\ldots,\phi_{k-1}\})$ and
- apply the the quotient $\mathrm{Stab}(\{\phi_0,\ldots,\phi_{k-1}\})/\,\mathrm{Stab}(T)\cap\mathrm{Stab}(\{\phi_0,\ldots,\phi_{k-1}\})$.

**Remark:** we obtain $\mathcal{C}_{r-\ell+k}(\mathrm{Span}(\phi_0,\ldots,\phi_{k-1}))/\,\mathrm{Stab}(T)\cap\mathrm{Stab}(\{\phi_0,\ldots,\phi_{k-1}\})$, slightly larger than the targeted set.

# Technical aspects

How to compute
$\mathcal{C}_{r-\ell+k}(\mathrm{Span}(\phi_0, \ldots, \phi_{k-1}))/\mathrm{Stab}(\{\phi_0, \ldots, \phi_{k-1}\})$?

**Algorithm:**

- for all $W \in \mathscr{S}_{r-\ell+k}/\mathrm{GL}_n \times GL_n$, enumerate all the tuples $(\psi_0, \ldots, \psi_{k-1})$ such that $\psi_i \in W$ and $\mathrm{rk}(\psi_i) = \mathrm{rk}(\phi_i)$;
- compute $\sigma \in \mathrm{GL}_n \times \mathrm{GL}_n$ such that $\{\psi_0, \ldots, \psi_{k-1}\} \circ \sigma = \{\phi_0, \ldots, \phi_{k-1}\}$ (computational group theory, Weierstrass-Kronecker theory...).

# Covering sets on examples

Covering set for the short product modulo $X^5$:

- $\mathcal{C}_8(\mathrm{Span}(c_3, c_4))$.

Covering sets for the product of matrices $3 \times 2$ by $2 \times 3$ (the coefficients are denoted by $c_{i,j}$):

- $\mathcal{C}_7(\mathrm{Span}(c_{0,0} + c_{1,1} + c_{2,2}))$;
- $\mathcal{C}_8(\mathrm{Span}(c_{0,0} + c_{1,1}, c_{0,0} + c_{2,2}))$;
- $\mathcal{C}_8(\mathrm{Span}(c_{0,0} + c_{1,1}, c_{0,1} + c_{2,2}))$;
- $\mathcal{C}_8(\mathrm{Span}(c_{0,0} + c_{1,1}, c_{2,2}))$;
- $\mathcal{C}_9(\mathrm{Span}(c_{0,0}, c_{1,1}, c_{2,2}))$.

# Timings

We have timings on a single core 3.3 GHz Intel Core i5-4590.

| product | time (s) | nb. of solutions |
|---|---|---|
| $\text{ShortProduct}_4$ | 3.0 | $1,440$ |
| $\text{ShortProduct}_5$ | $2.4 \cdot 10^3$ | $146,944$ |

Table: Computation of decompositions of the short product.

| product | time (s) | nb. of solutions |
|---|---|---|
| $2 \times 3$ by $3 \times 2$ | $4.1 \cdot 10^6$ | $1,096,452$ |
| $3 \times 2$ by $2 \times 3$ | $3.0 \cdot 10^6$ | $7,056$ |

Table: Computation of decompositions of the matrix product.

# Conclusion

We obtain interesting speed-up for symmetric bilinear maps such as matrix product and short product compared to implementations of BDEZ.

What kind of covering sets for product of polynomials (small group of symmetry)?

How to push computations further: possible to decompose matrix product $3 \times 3$ by $3 \times 3$?