Improving Thomae-Wolf Algorithm for Solving Underdetermined Multivariate Quadratic Polynomial Problem

Hiroki Furue[1], Shuhei Nakamura[2], Tsuyoshi Takagi[1]

1. The University of Tokyo, Japan
2. Nihon University, Japan

PQCrypto 2021

Our Contributions

MQ problem of m equations in n variables over $\mathbb{F}_{2^r}$

$n > m$ (underdetermined)

k : the number of guessed variables

Algorithm	Resulting system	
	Variables	Equations
Hybrid approach	$m - k$	m
Hybrid + Thomae-Wolf	$m - \left(\left\lfloor \frac{n}{m} \right\rfloor - 1\right) - k$	$m - \left(\left\lfloor \frac{n}{m} \right\rfloor - 1\right)$
Our algorithm	$m - \left(\left\lfloor \frac{n-k}{m-k} \right\rfloor - 1\right) - k$	$m - \left(\left\lfloor \frac{n-k}{m-k} \right\rfloor - 1\right)$
Our algorithm ($\mathbb{F}_2$)	$m - \left(\left\lfloor \frac{n-1}{m-k-1} \right\rfloor - 1\right) - k$	$m - \left(\left\lfloor \frac{n-1}{m-k-1} \right\rfloor - 1\right)$

Outline

- MQ problem
- Thomae-Wolf Algorithm
- Proposed Algorithm
- Proposed Algorithm for the Binary Field
- Conclusion

Post Quantum Cryptography

We need cryptosystems secure against quantum computers.

- Multivariate polynomial cryptography
- Lattice-based cryptography
- Code-based cryptography
- Hash-based cryptography
- Isogeny-based cryptography

MQ Problem

$MQ(q, n, m)$

- $\mathbb{F}_q$: Finite field of order q
- n : the number of variables
- m : the number of equations

$$\sum_{i \le j} a_{ij}^{(1)} x_i x_j + \sum_i b_i^{(1)} x_i + c^{(1)} = 0$$

$$\vdots$$

$$\sum_{i \le j} a_{ij}^{(m)} x_i x_j + \sum_i b_i^{(m)} x_i + c^{(m)} = 0$$

$$\left(a_{ij}^{(k)}, b_i^{(k)}, c^{(k)} \in \mathbb{F}_q \right)$$

Solving MQ problem

Hybrid Approach

[Yang et al. ICICS 2004]
[Bettale et al., J. Mathematical Cryptology, 2009]

$k \in \{1, \dots, m\}$

① fix k variables $x_{n-k+1}, \dots, x_n$ randomly

② solve the resulting $MQ(q, n-k, m)$ (by using F4, F5, XL)

※ repeat ①, ② until a solution is obtained

- In the case of $n > m$ (underdetermined)

If we fix $n - m$ variables $x_{m+1}, \dots, x_n$ randomly,
then there exists a solution with high probability.

$MQ(q, n, m) \Rightarrow MQ(q, m, m)$

Outline

- MQ problem
- Thomae-Wolf Algorithm
- Proposed Algorithm
- Proposed Algorithm for the Binary Field
- Conclusion

Thomae-Wolf Algorithm

[Thomae, Wolf, PKC 2012]

$\mathcal{P} = (p_1, \dots, p_m)$: Underdetermined MQ system

Solve $\mathcal{P}(x_1, \dots, x_n) = \mathbf{0}$

$$p_\ell = \sum_{i \le j} a_{ij}^{(\ell)} x_i x_j + \sum_i b_i^{(\ell)} x_i + c^{(\ell)}$$

Introduce a linear map $S = (\boldsymbol{s}_1 \cdots \boldsymbol{s}_n)$
s.t. $\mathcal{F} \coloneqq \mathcal{P} \circ S$ has a special structure

$$f_\ell = \sum_{i \le j} \bar{a}_{ij}^{(\ell)} x_i x_j + \sum_i \bar{b}_i^{(\ell)} x_i + \bar{c}^{(\ell)}$$

$\boxed{\bar{a}_{ij}^{(\ell)} \text{ depends on } \boldsymbol{s}_i \text{ and } \boldsymbol{s}_j}$

Thomae-Wolf: Step 1

α: linearization factor $(1 \leq \alpha \leq m)$

(1-1) fix $\boldsymbol{s}_1$ randomly

(1-2) solve $\bar{a}_{12}^{(\ell)} = 0 \;\; (1 \leq \ell \leq \alpha)$ for $\boldsymbol{s}_2$

(1-3) solve $\bar{a}_{13}^{(\ell)} = 0$

$$\bar{a}_{23}^{(\ell)} = 0 \;\; (1 \leq \ell \leq \alpha) \text{ for } \boldsymbol{s}_3$$

$\vdots$

(1- m) solve $\bar{a}_{1m}^{(\ell)} = 0$

$$\vdots$$

$$\bar{a}_{(m-1)m}^{(\ell)} = 0 \;\; (1 \leq \ell \leq \alpha) \text{ for } \boldsymbol{s}_m$$

Thomae-Wolf: Step 1

The resulting system $(f_1, \dots, f_m)$

$$f_1 = \sum_{i=1}^{m} \bar{a}_{ii}^{(1)} x_i^2 + \sum_{i=1}^{m} x_i \underbrace{L_i^{(1)}(x_{m+1}, \dots, x_n)}_{\text{linear}} + \underbrace{Q^{(1)}(x_{m+1}, \dots, x_n)}_{\text{quadratic}}$$

$$\vdots$$

$$f_\alpha = \sum_{i=1}^{m} \bar{a}_{ii}^{(\alpha)} x_i^2 + \sum_{i=1}^{m} x_i L_i^{(\alpha)}(x_{m+1}, \dots, x_n) + Q^{(\alpha)}(x_{m+1}, \dots, x_n)$$

$$f_{\alpha+1} = Q^{(\alpha+1)}(x_1, \dots, x_n)$$

$$\vdots$$

$$f_m = Q^{(m)}(x_1, \dots, x_n)$$

Thomae-Wolf: Step 2

$$f_1 = \sum_{i=1}^{m} \bar{a}_{ii}^{(1)} x_i^2 + \sum_{i=1}^{m} x_i L_i^{(1)}(x_{m+1}, \ldots, x_n) + Q^{(1)}(x_{m+1}, \ldots, x_n)$$

$$\vdots$$

$$f_\alpha = \sum_{i=1}^{m} \bar{a}_{ii}^{(\alpha)} x_i^2 + \sum_{i=1}^{m} x_i L_i^{(\alpha)}(x_{m+1}, \ldots, x_n) + Q^{(\alpha)}(x_{m+1}, \ldots, x_n)$$

Solve

$$L_i^{(\ell)}(x_{m+1}, \ldots, x_n) = 0 \quad (i \in \{1, \ldots, m\}, \ell \in \{1, \ldots, \alpha\})$$

for $(x_{m+1}, \ldots, x_n)$

$\Rightarrow$ $(n - m)$ variables, αm equations

Thomae-Wolf: Step 3

Substitute the values obtained in Step 2 for $(x_{m+1}, \dots, x_n)$

$$\left.\begin{array}{l} \sum_{i=1}^{m} \bar{a}_{ii}^{(1)} x_i^2 + c'^{(1)} = 0 \\ \vdots \\ \sum_{i=1}^{m} \bar{a}_{ii}^{(\alpha)} x_i^2 + c'^{(\alpha)} = 0 \end{array}\right\}$$

$$Q'^{(\alpha+1)}(x_1, \dots, x_m) = 0$$

$$\vdots$$

$$Q'^{(m)}(x_1, \dots, x_m) = 0$$

In the case $\mathbb{F}_{2^r}$,
rase to the 2^{r-1}-th power

$$\sum_{i=1}^{m} \left(\bar{a}_{ii}^{(1)}\right)^{2^{r-1}} x_i + c''^{(1)} = 0$$

$$\vdots$$

$$\sum_{i=1}^{m} \left(\bar{a}_{ii}^{(\alpha)}\right)^{2^{r-1}} x_i + c''^{(\alpha)} = 0$$

By using these equations,

we obtain $MQ(2^r, m - \alpha, m - \alpha)$

Thomae-Wolf: α

Step 1- i : $(n-m)$ variables, $\alpha(i-1)$ equations

$(i = 2, \ldots, m)$

Step 2 : $(n-m)$ variables, αm equations

To obtain the solutions of these systems,

$$n - m \geq \alpha m$$

$$\alpha \leq \frac{n}{m} - 1$$

$$\therefore \alpha = \left\lfloor \frac{n}{m} \right\rfloor - 1$$

Hybrid + Thomae-Wolf

$MQ(2^r, n, m)\ \ (n > m)$

① Thomae-Wolf Algorithm

$$\Rightarrow\ MQ\left(2^r, m - \left(\left\lfloor \frac{n}{m} \right\rfloor - 1\right), m - \left(\left\lfloor \frac{n}{m} \right\rfloor - 1\right)\right)$$

② Hybrid Approach

$$\Rightarrow\ MQ\left(2^r, m - \left(\left\lfloor \frac{n}{m} \right\rfloor - 1\right) - k, m - \left(\left\lfloor \frac{n}{m} \right\rfloor - 1\right)\right)$$

(for each k guessed variables)

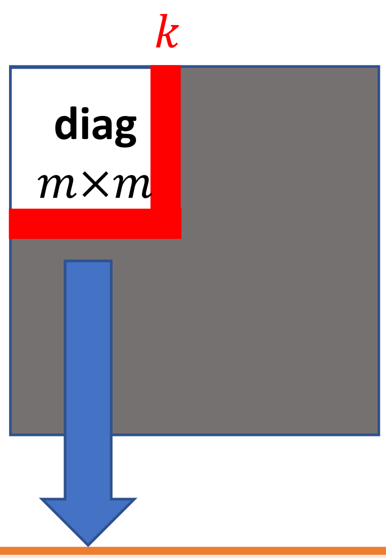
Outline

- MQ problem
- Thomae-Wolf Algorithm
- Proposed Algorithm
- Proposed Algorithm for the Binary Field
- Conclusion

Proposed Algorithm: Idea

In the Thomae-Wolf algorithm,

the representation matrix of f_i $(1 \leq i \leq \alpha)$:



We can omit the structure corresponding k variables fixed in the hybrid approach.

Proposed Algorithm: Step 1

α_k : linearization factor $(1 \leq \alpha_k \leq m)$

(1-1) fix $\boldsymbol{s}_1$ randomly

(1-2) solve $\bar{a}_{12}^{(\ell)} = 0 \ \ (1 \leq \ell \leq \alpha_k)$ for $\boldsymbol{s}_2$

(1-3) solve $\bar{a}_{13}^{(\ell)} = 0$

$\bar{a}_{23}^{(\ell)} = 0 \ \ (1 \leq \ell \leq \alpha_k)$ for $\boldsymbol{s}_3$

$\vdots$

(1- $(m-k)$) solve $\bar{a}_{1(m-k)}^{(\ell)} = 0$

$\vdots$

$\bar{a}_{(m-k-1)(m-k)}^{(\ell)} = 0 \ \ (1 \leq \ell \leq \alpha_k)$ for $\boldsymbol{s}_{m-k}$

Proposed Algorithm: Step 1

The resulting system $(f_1, \dots, f_m)$

$$f_1 = \sum_{i=1}^{m-k} \bar{a}_{ii}^{(1)} x_i^2 + \sum_{i=1}^{m-k} x_i \underbrace{L_i^{(1)}(x_{m-k+1}, \dots, x_n)}_{\text{linear}} + \underbrace{Q^{(1)}(x_{m-k+1}, \dots, x_n)}_{\text{quadratic}}$$

$$\vdots$$

$$f_{\alpha_k} = \sum_{i=1}^{m-k} \bar{a}_{ii}^{(\alpha_k)} x_i^2 + \sum_{i=1}^{m-k} x_i L_i^{(\alpha_k)}(x_{m-k+1}, \dots, x_n) + Q^{(\alpha_k)}(x_{m-k+1}, \dots, x_n)$$

$$f_{\alpha_k+1} = Q^{(\alpha_k+1)}(x_1, \dots, x_n)$$

$$\vdots$$

$$f_m = Q^{(m)}(x_1, \dots, x_n)$$

Proposed Algorithm: Step 2

$$f_1 = \sum_{i=1}^{m-k} \bar{a}_{ii}^{(1)} x_i^2 + \sum_{i=1}^{m-k} x_i L_i^{(1)}(x_{m-k+1}, \dots, x_n) + Q^{(1)}(x_{m-k+1}, \dots, x_n)$$

$$\vdots$$

$$f_{\alpha_k} = \sum_{i=1}^{m-k} \bar{a}_{ii}^{(\alpha_k)} x_i^2 + \sum_{i=1}^{m-k} x_i L_i^{(\alpha_k)}(x_{m-k+1}, \dots, x_n) + Q^{(\alpha_k)}(x_{m-k+1}, \dots, x_n)$$

Solve

$$L_i^{(\ell)}(x_{m-k+1}, \dots, x_n) = 0 \ \ (i \in \{1, \dots, m-k\}, \ell \in \{1, \dots, \alpha_k\})$$

only for $(x_{m+1}, \dots, x_n)$

$$\Rightarrow \quad \begin{aligned} x_{m+1} &= L'_{m+1}(x_{m-k+1}, \dots, x_m) \\ &\vdots \\ x_n &= L'_n(x_{m-k+1}, \dots, x_m) \end{aligned}$$

Proposed Algorithm: Step 3

Fix $(x_{m-k+1}, \dots, x_m) = (c_{m-k+1}, \dots, c_m)$ randomly

Substitute $(x_{m-k+1}, \dots, x_n) =$
$(c_{m-k+1}, \dots, c_m, L'_{m+1}(c_{m-k+1}, \dots, c_m), \dots, L'_n(c_{m-k+1}, \dots, c_m))$

$$\sum_{i=1}^{m-k} \bar{a}_{ii}^{(1)} x_i^2 + c'^{(1)} = 0$$

$$\vdots$$

$$\sum_{i=1}^{m-k} \bar{a}_{ii}^{(\alpha_k)} x_i^2 + c'^{(\alpha_k)} = 0$$

$$Q'^{(\alpha_k+1)}(x_1, \dots, x_{m-k}) = 0$$

$$\vdots$$

$$Q'^{(m)}(x_1, \dots, x_{m-k}) = 0$$

After that, use the same method as in the Thomae-Wolf algorithm.

$MQ(2^r, n, m)$

$\Rightarrow MQ(2^r, m - \alpha_k - k, m - \alpha_k)$

Proposed Algorithm: α_k

Step 1- i : $\left(n-(m-k)\right)$ variables, $\alpha_k(i-1)$ equations

$(i=2,\ldots,m-k)$

Step 2 : $(n-m)$ variables, $\alpha_k(m-k)$ equations

To obtain the solutions of these systems,

$$n-m \geq \alpha_k(m-k)$$

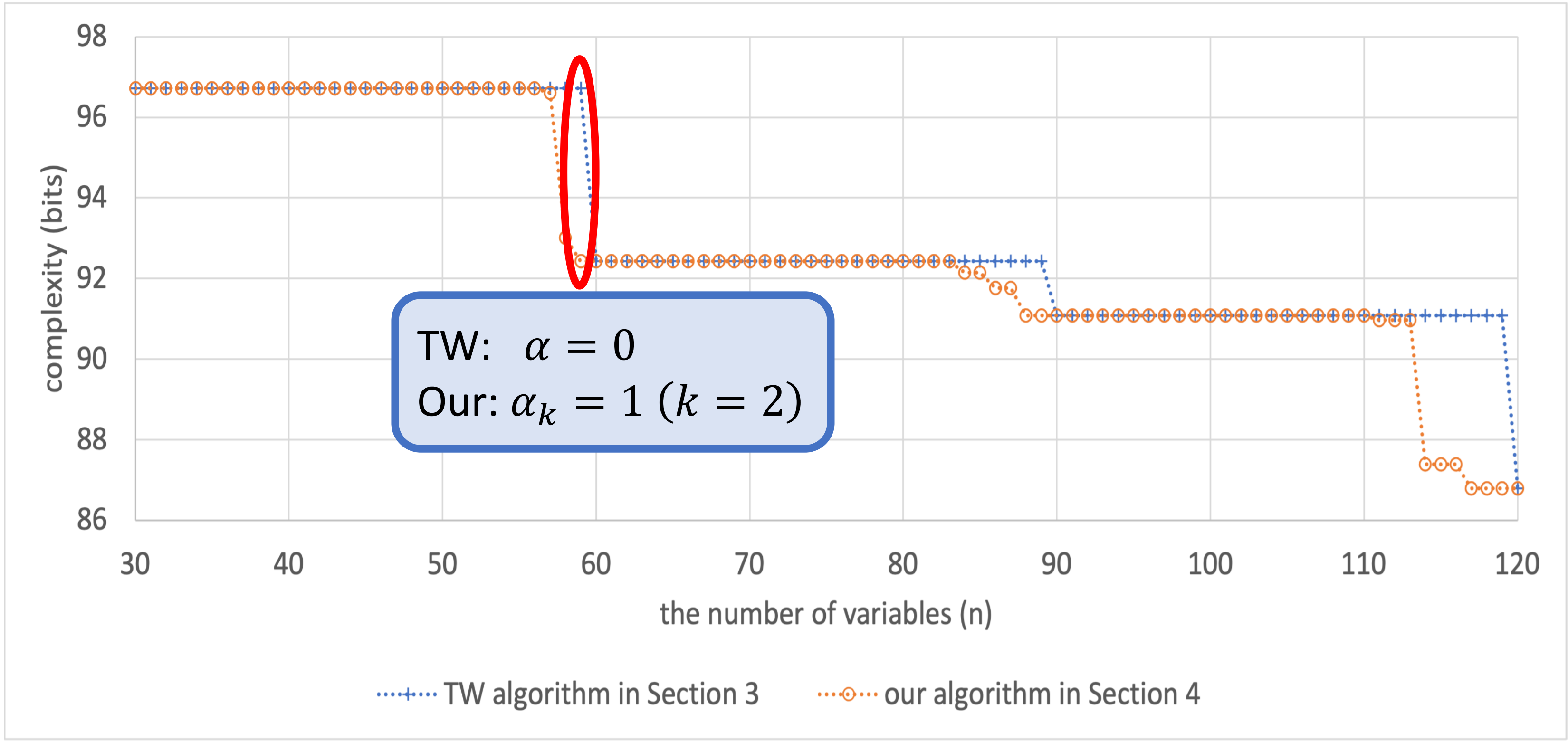
$$\alpha_k \leq \frac{n-m}{m-k} = \frac{n-k}{m-k} - 1$$

$$\therefore \alpha_k = \left\lfloor \frac{n-k}{m-k} \right\rfloor - 1$$

(Thomae-Wolf algorithm: $\alpha = \left\lfloor \frac{n}{m} \right\rfloor - 1$)

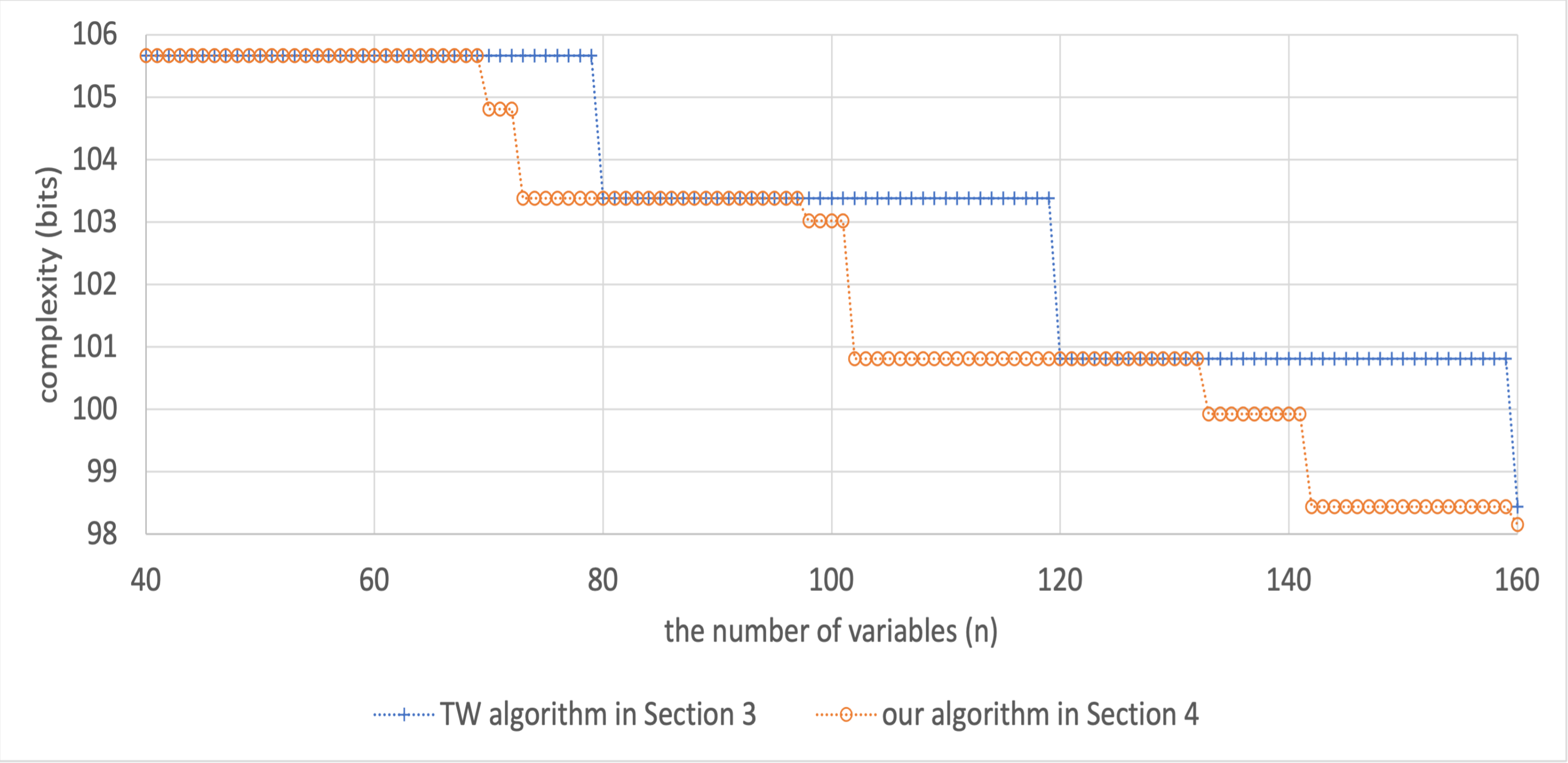
Theoretical Complexity

$q = 2^8, m = 30, 30 \leq n \leq 120$



Theoretical Complexity

$q = 2^4, m = 40, 40 \leq n \leq 160$



Outline

- MQ problem
- Thomae-Wolf Algorithm
- Proposed Algorithm
- Proposed Algorithm for the Binary Field
- Conclusion

Proposed Algorithm for $\mathbb{F}_2$

β_k : linearization factor ($1 \leq \beta_k \leq m$)

Step 1: same as the proposed algorithm for $\mathbb{F}_{2^r}$

$$f_\ell = \sum_{i=1}^{m-k} \bar{a}_{ii}^{(\ell)} x_i^2 + \sum_{i=1}^{m-k} x_i L_i^{(\ell)}(x_{m-k+1}, \dots, x_n) + Q^{(\ell)}(x_{m-k+1}, \dots, x_n)$$

$$(1 \leq \ell \leq \beta_k)$$

$x_i^2 = x_i \ (\mathbb{F}_2)$

$$f_\ell = \sum_{i=1}^{m-k} x_i L_i^{(\ell)}(x_{m-k+1}, \dots, x_n) + Q^{(\ell)}(x_{m-k+1}, \dots, x_n)$$

We can omit Step 2

Step 3: same as the proposed algorithm for $\mathbb{F}_{2^r}$

Proposed Algorithm for $\mathbb{F}_2$

Step 1- i : $(n-(m-k))$ variables, $\beta_k(i-1)$ equations

$(i = 2, \ldots, m-k)$

Step 2 : $(n-m)$ variables, $\beta_k(m-k)$ equations

To obtain the solutions of these systems,

$$n-(m-k) \geq \beta_k(m-k-1)$$

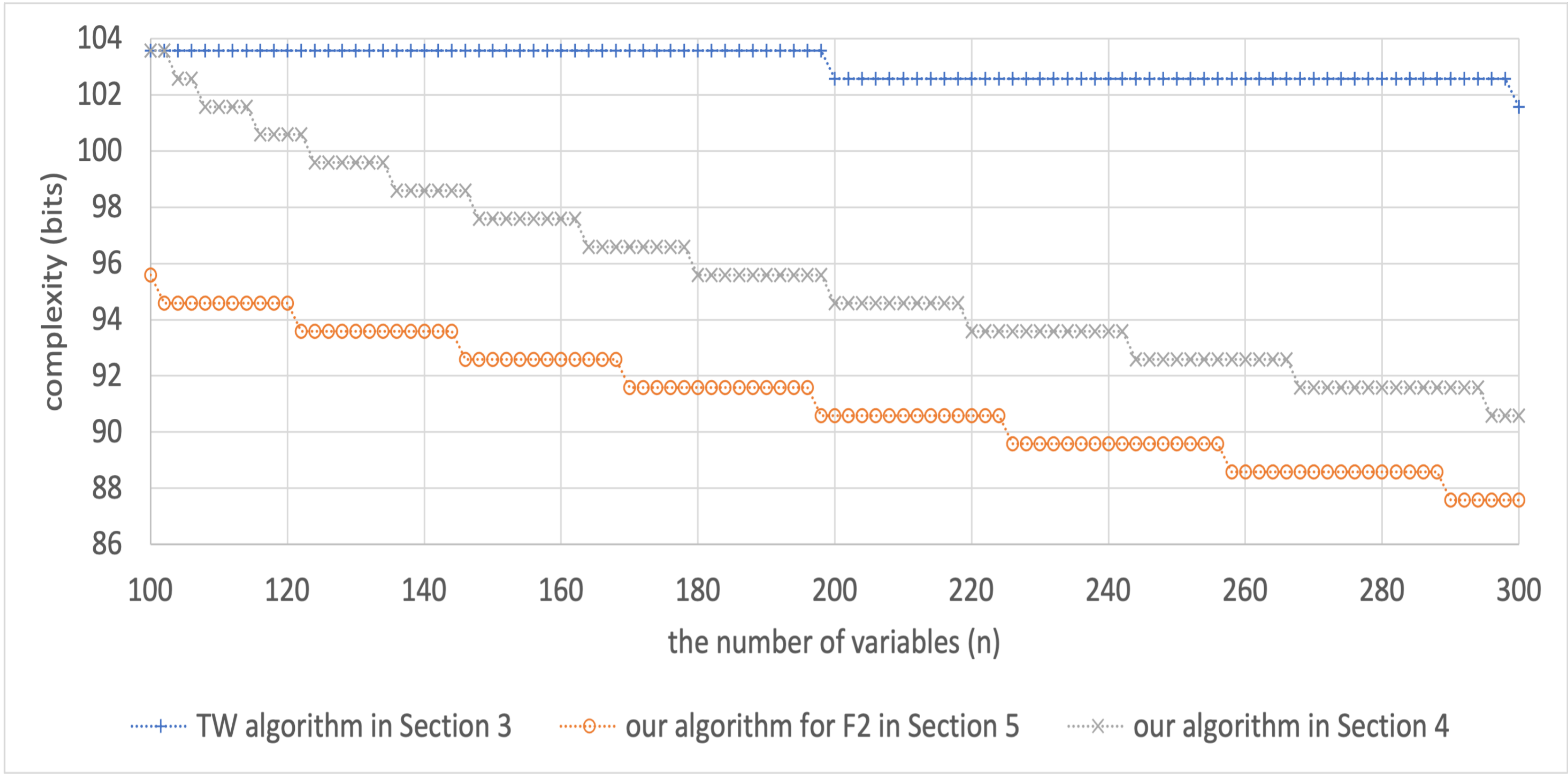
$$\beta_k \leq \frac{n-(m-k)}{m-k-1} = \frac{n-1}{m-k-1} - 1$$

$$\therefore \beta_k = \left\lfloor \frac{n-1}{m-k-1} \right\rfloor - 1$$

(the proposed algorithm for $\mathbb{F}_{2^r}$: $a_k = \left\lfloor \frac{n-k}{m-k} \right\rfloor - 1$)

Theoretical complexity

$q = 2, m = 100, 100 \leq n \leq 300$



Outline

- MQ problem
- Thomae-Wolf Algorithm
- Proposed Algorithm
- Proposed Algorithm for the Binary Field
- Conclusion

Conclusion

- For the underdetermined MQ problem,
  we proposed a new efficient algorithm
  by improving the Thomae-Wolf algorithm.

- In future work, we will consider the application of
  the proposed algorithm for $\mathbb{F}_2$ to existing algorithms.