Small Public Keys and Fast Verification for Multivariate Quadratic Public Key Systems

TECHNISCHE UNIVERSITÄT DARMSTADT

Albrecht Petzoldt[1], Enrico Thomae[2], Stanislav Bulygin[3] and Christopher Wolf[4]

[1,3]Technische Universität Darmstadt, CASED
[2,4]Ruhr-Universität Bochum

Outline

Motivation

The UOV Signature Scheme

Review: Reducing public key size

„Security proof“ of the Construction

The new approach: 0/1 UOV

Parameters and Implementation

Conclusion and Future Work

Our Contribution

Multivariate Cryptography

- Candidate for Post-Quantum Cryptography

- Low computational requirements
- Fast and efficient

- Large key sizes
- Security ?

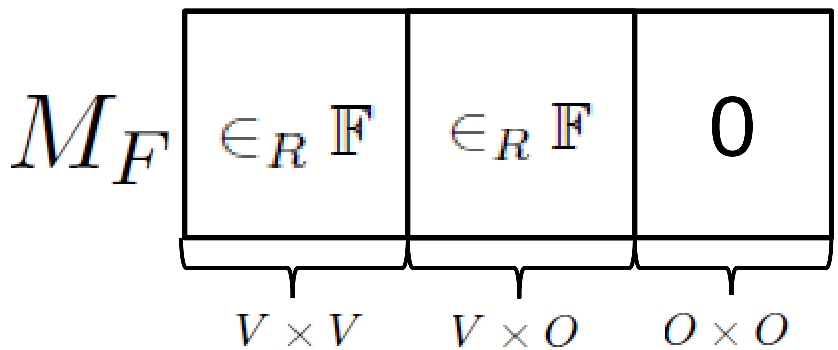
The Oil and Vinegar Signature Scheme

TECHNISCHE UNIVERSITÄT DARMSTADT

Two types of variables: Oil and Vinegar

- Central map $\mathcal{F}$ of o quadratic polynomials of the form

$$f^{(k)}(u_1,\ldots,u_n) = \sum_{i,j\in V,\ i\leq j} f_{ij}^{(k)} u_i u_j + \sum_{i\in V,\ j\in O} f_{ij}^{(k)} u_i u_j \quad (k=1,\ldots,o)$$



- linear invertible map $\mathcal{S}$

public key: $\mathcal{P} = \mathcal{F} \circ \mathcal{S}$

private key: $\mathcal{F},\ \mathcal{S}$

Oil and Vinegar (2)

Signature generation

- Compute $\mathbf{h} = \mathcal{H}(m) \in \mathbb{F}^o$
- Compute one preimage of $\mathbf{h}$ under $\mathcal{F}$
  - Assign random values to the Vinegar variables $u_1, \ldots, u_v$
  - Solve the resulting linear system for the Oil variables $u_{v+1}, \ldots, u_n$
- Compute $\mathbf{x} = \mathcal{S}^{-1}(\mathbf{u}) \in \mathbb{F}^n$

Signature verification

- Compute $\mathbf{h} = \mathcal{H}(m)$ and $\mathbf{h}' = \mathcal{P}(\mathbf{x})$.
- $\mathbf{h}' = \mathbf{h}$ → accept the signature
  else reject

Recommended Parameters: $(q,o,v)=(2^8,26,52)$

Reducing public key size

M_P

103	172	182	091	165	207	143	125	173	072	163	174	183	195
173	093	248	183	076	172	152	251	125	179	082	238	193	078
182	235	196	083	102	186	112	241	139	087	118	241	156	207
193	229	051	213	194	146	173	247	072	184	239	092	173	274
153	242	097	162	252	183	089	173	218	138	243	158	142	093

Reducing public key size

The approach of PB10

$$D := \frac{v \cdot (v+1)}{2} + o \cdot v$$

$$M_P = \underbrace{\begin{array}{cccccccc} c_1 & c_2 & c_3 & c_4 & \cdots & c_{D-2} & c_{D-1} & c_D \\ c_D & c_1 & c_2 & c_3 & \cdots & c_{D-3} & c_{D-2} & c_{D-1} \\ c_{D-1} & c_D & c_1 & c_2 & \cdots & c_{D-4} & c_{D-3} & c_{D-2} \\ \vdots & & & & \ddots & & & \vdots \end{array}}_{B} \; \underbrace{\begin{array}{c} 103\ 172\ 182\ 091 \\ 173\ 072\ 163\ 174 \\ 248\ 183\ 076\ 172 \\ 152\ 251\ 125\ 179 \\ 082\ 238\ 193\ 078 \end{array}}_{C}$$

78.2 kB → 11.2 kB

→ Key size reduction by up to 85 %

The approach of PB10

TECHNISCHE UNIVERSITÄT DARMSTADT

Observation

$$p^{(k)} : \sum_{i=1}^{n} \sum_{j=i}^{n} p_{ij}^{(k)} x_i x_j$$

$$f^{(k)} : \sum_{r=1}^{v} \sum_{s=r}^{n} f_{rs}^{(k)} u_r u_s$$

$$\mathcal{P} = \mathcal{F} \circ \mathcal{S} \quad \Longrightarrow \quad p_{ij}^{(k)} = \sum_{r=1}^{v} \sum_{s=r}^{n} \alpha_{ij}^{rs} \cdot f_{rs}^{(k)}$$

with

$$\alpha_{ij}^{rs} = \begin{cases} s_{ri} \cdot s_{si} & (i = j) \\ s_{ri} \cdot s_{sj} + s_{rj} \cdot s_{si} & \text{otherwise} \end{cases}$$

The approach of PB10

TECHNISCHE UNIVERSITÄT DARMSTADT

Set $D := \frac{v \cdot (v+1)}{2} + o \cdot v$

- Choose an o x D matrix B
- Choose randomly the linear invertible map $\mathcal{S}$.

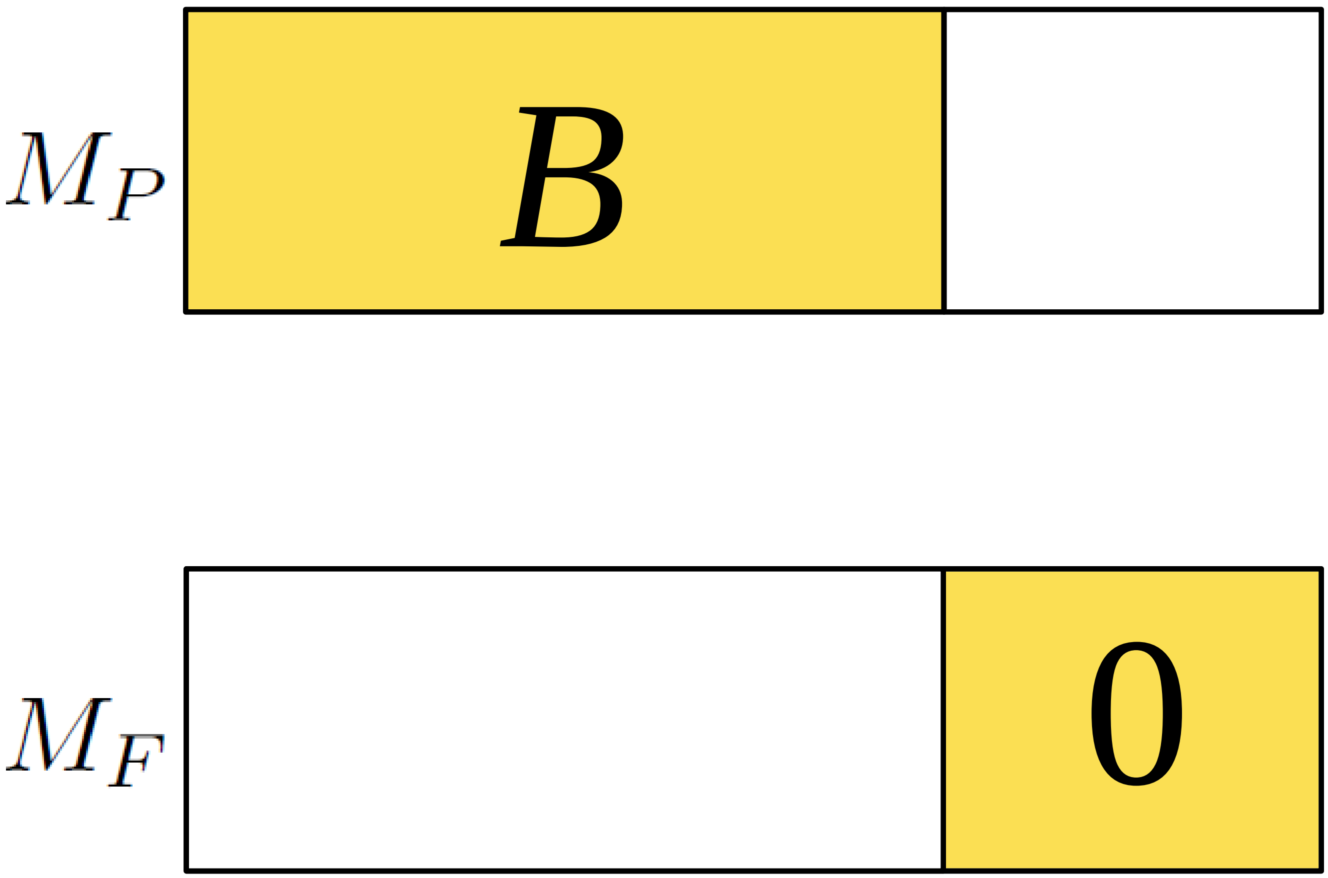
Compute for $\mathcal{S}$ the D x D transformation matrix

$$A = \begin{pmatrix} \alpha_{11}^{11} & \alpha_{12}^{11} & \dots & \alpha_{vn}^{11} \\ \alpha_{11}^{12} & & & \alpha_{vn}^{12} \\ \vdots & & & \vdots \\ \alpha_{11}^{vn} & \alpha_{12}^{vn} & \dots & \alpha_{vn}^{vn} \end{pmatrix}$$

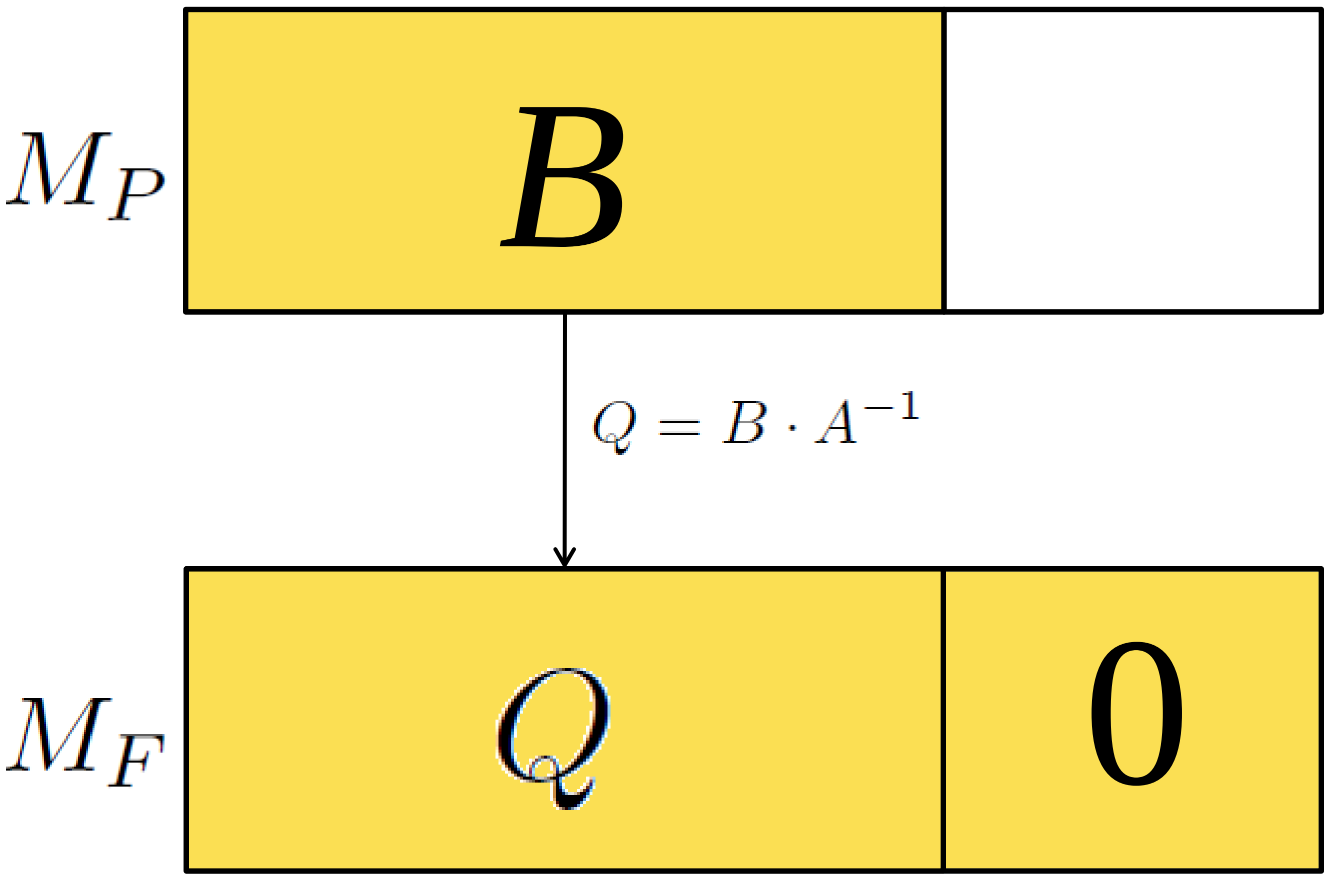
where

$$\alpha_{ij}^{rs} = \begin{cases} s_{ri} \cdot s_{si} & (i = j) \\ s_{ri} \cdot s_{sj} + s_{rj} \cdot s_{si} & \text{otherwise} \end{cases}$$

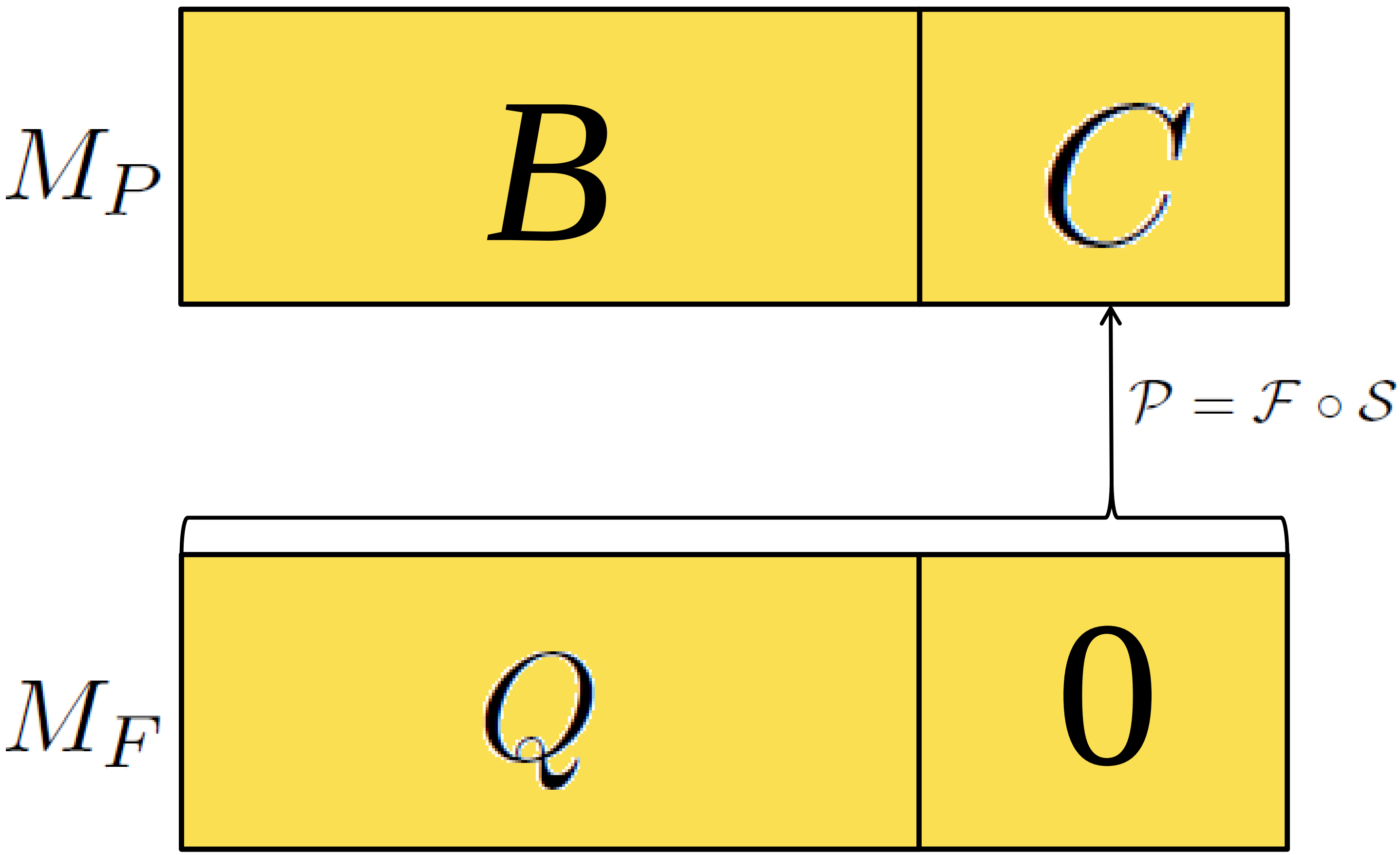
The approach of PB10



The approach of PB10

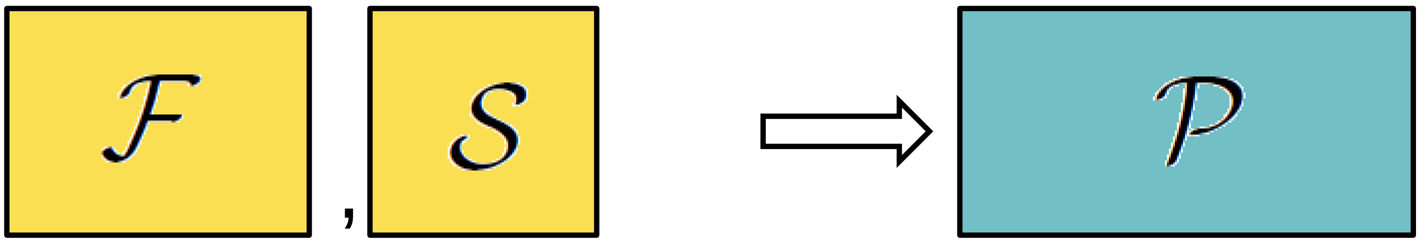


The approach of PB10

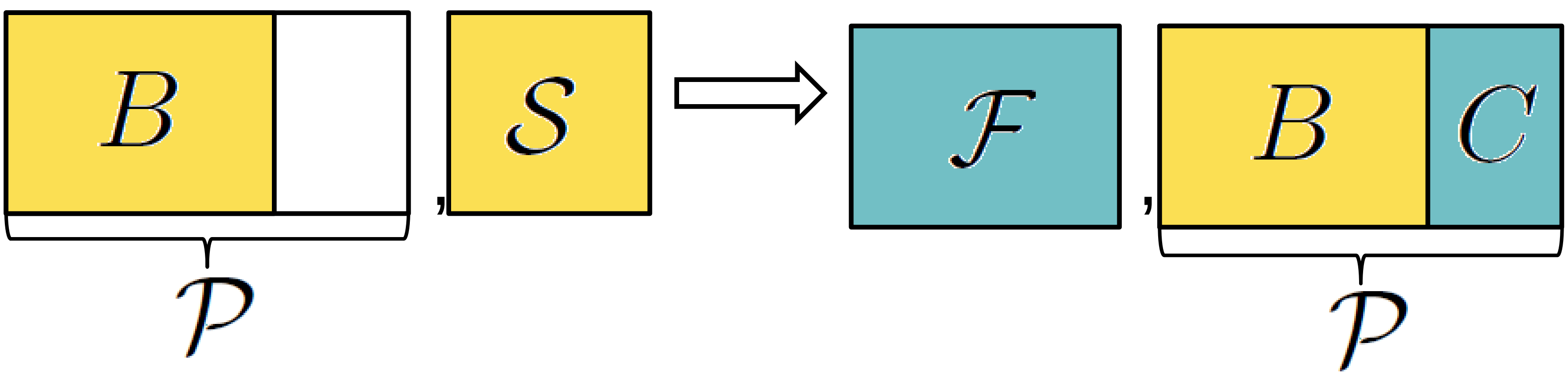


The approach of PB10

Standard Construction

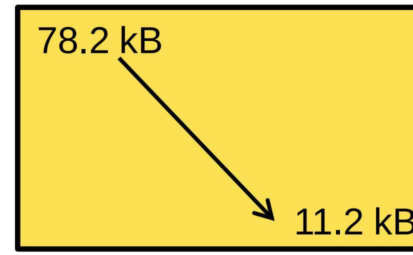


New Construction



Result of PB10

Reduction of the public key size by up to 85 %



But: What about the security?

Security

Proposition: Let B an MDS matrix. Then, in the sense of key recovery attacks, the new construction is as secure as the standard key generation of UOV.

Equivalent keys

Let $(\mathcal{F}, \mathcal{S})$ and $(\mathcal{F}', \mathcal{S}')$ be two UOV private keys. They are called equivalent iff they result in the same public key, i.e.

$$\mathcal{F} \circ \mathcal{S} = \mathcal{F}' \circ \mathcal{S}' =: \mathcal{P}$$

Security (2)

Lemma: For each UOV public key $\mathcal{P}$ there exists a UOV private key $(\widetilde{F}, \widetilde{S})$ s. t. $\widetilde{S}$ has the form

$$\widetilde{S} = \begin{pmatrix} I_{v \times v} & \widetilde{S}'_{v \times o} \\ 0_{o \times v} & I_{o \times o} \end{pmatrix}$$

Lemma: For each UOV public key $\mathcal{P}$ there exists a UOV private key $(\widetilde{F}, \widetilde{S})$ such that

$$\widetilde{f_{ij}^{(k)}} = p_{ij}^{(k)} \; \forall k \in \{1, \ldots, o\}, \; i, j \in \{1, \ldots, v\}$$.

What we have now

Reduction of the public key size by up to 85 %

78.2 kB → 11.2 kB

+ „Security proof“

Can we do even better than PB10?

- in terms of public key size
- in terms of verification cost

Idea: Use a matrix B defined over GF(2)

The new approach: 0 / 1 UOV

M_P

1 0 0 1 0 1 0 0 1 1 0 1 1 0 0 1 1 0 1 0 1 1 0 1 0 1	103 172 182 091
0 1 1 0 1 0 1 0 0 1 0 1 1 0 0 1 0 1 1 1 0 0 1 1 0 0	173 072 163 174
1 0 1 1 0 1 1 0 1 0 1 0 1 1 0 1 0 0 1 1 0 0 0 1 0 1	248 183 076 172
0 1 0 1 0 1 0 0 1 0 1 0 1 1 0 0 1 0 1 1 1 0 1 0 1 1	152 251 125 179
1 1 0 0 1 0 1 0 1 0 1 1 0 0 0 1 0 1 0 1 1 0 1 0 1 0	082 238 193 078
B	C

- **Problem: Direct attacks**

By fixing some variables an attacker might be able to turn all the monomials over GF(2^8) into constants

→he could compute a Gröbner basis over GF(2)

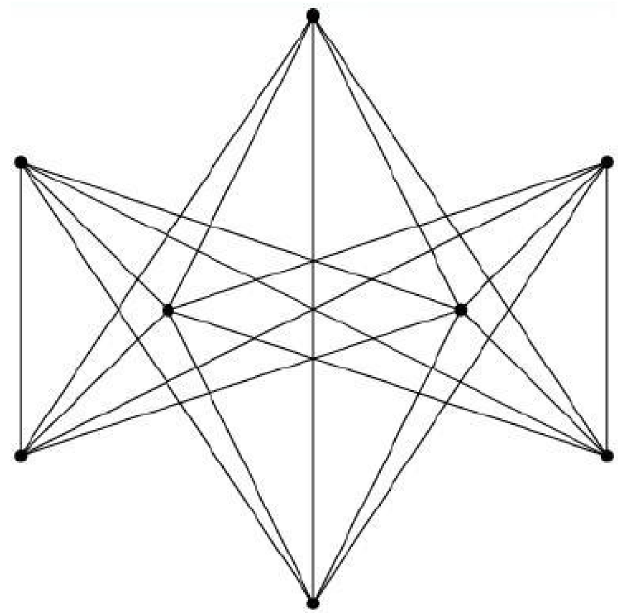
- **Solution**: Use another ordering of monomials

The Turán graph $T(n,k)$

- Divide the set $V = \{v_1, \ldots, v_n\}$ of vertices into k subsets $A_i \ (i = 1, \ldots, k)$.

$$A_i \cap A_j = \emptyset,\ ||A_i| - |A_j|| \leq 1 \ (i \neq j)$$

- Two vertices are connected by an edge iff they belong to different subsets



T(8,3)

Theorem: The Turán graph $T(n,k)$ is the graph with the maximal number of edges which does not contain a (k+1)-clique, i.e.

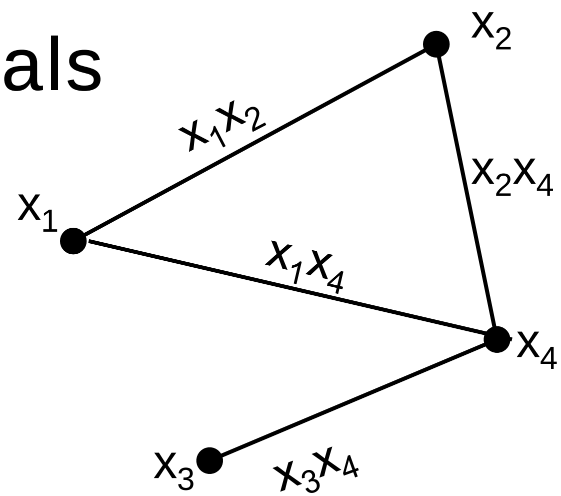
$$\nexists V' \subset V \text{ with } |V'| = k+1 \text{ s.t. } \{e(v_i, v_j) : v_i, v_j \in V'\} \subset E$$

0 / 1 UOV

TECHNISCHE UNIVERSITÄT DARMSTADT

Graph ↔ Ordering of monomials

- Vertices ↔ variables
- Edges ↔ quadratic monomials



3 Blocks:

1. Squared variables (e.g. x_1^2)
2. Monomials represented by edges of the graph
3. Remaining monomials

- Inside the blocks we use the lexicographic order

→ use an ordering of monomials induced by the Turán graph.

Result

M_P = [B | C]

squared variables (B)	edges of $T(n,k)$ (B)	edges of $\overline{T}(n,k)$ (C)
1 0 0 1 0 1 0	0 1 1 0 1 1 0 0 1 1 0 1 0 1 1 0 1 0 1	103 172 182 091
0 1 1 0 1 0 1	0 0 1 0 1 1 0 0 1 0 1 1 1 0 0 1 1 0 0	173 072 163 174
0 1 0 0 0 1 1	0 1 0 1 0 1 1 0 1 0 0 1 1 0 0 0 1 0 1	248 183 076 172
0 1 1 0 1 1 0	0 1 0 1 0 1 1 0 0 1 0 1 1 1 0 1 0 1 1	152 251 125 179
0 0 1 1 1 0 1	0 1 0 1 1 0 0 0 1 0 1 0 1 1 0 1 0 1 0	082 238 193 078

0 / 1 UOV

Direct Attacks

Before applying XL or a Gröbner Basis algorithm the attacker fixes/guesses at some variables to get an (over)determined system.

For $(q,o,v)=(2^8,26,52)$ there remain

- after fixing v variables at least 30 monomials with coefficients over $GF(2^8)$
- after fixing/guessing v+2=54 variables at least 24 monomials with coefficients over $GF(2^8)$

→ the attacker is not able to compute a Gröbner basis over GF(2).

Security of 0/1 UOV

- Security proof does not apply

→test the behaviour of known attacks against 0/1 UOV

  - Direct attacks
  - Rank attacks
  - UOV-Reconciliation attack
  - UOV attack

→ Known attacks cannot use the special structure of our public keys

Parameters

Recommended Parameters $(q,o,v)=(2^8,26,52)$.

Scheme (q,o,v)	System parameter (kB)	Private key size (kB)	Public key size (kB)	Reduction of public key size
UOV(2^8,26,52)	-	75.3	78.2	-
0/1 UOV(2^8,26,52)	8.7	75.3	8.9	88.6 %
UOV(2^8,28,56)	-	93.4	97.6	-
0/1 UOV(2^8,28,56)	10.8	93.4	11.1	88.6 %

Implementation

Key generation

- Computationally expensive
- we use M4RIE library and Travolta tables
- Running time on an Intel Dual Core 2.7 GHz ~27 sec

Signature Generation

- As for the standard UOV scheme: ~3.5 ms

Implementation (2)

Signature Verification (≈Evaluation of $\mathcal{P}$)

- Compute the values of all monomials $x_i x_j$ in advance → vector *mon*
- Compute for $i = 1,\ldots,o$ the scalar product $M_P[i] \cdot mon$
- elements of B ($\in GF(2)$)
  - If 1, carry out one addition
  - If 0, don't do anything

  B fixed → no need to perform if-clauses
- elements of C ($\in GF(2^8)$) → one multiplication + one addition

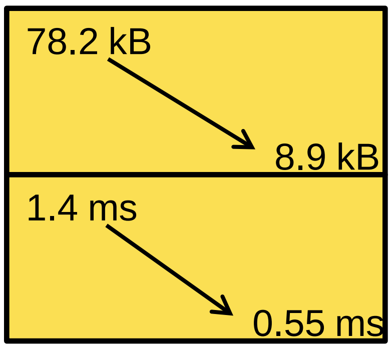
→ Reduction of the number of multiplications by 86 %

(q,o,v)	UOV	0 / 1 UOV	Reduction factor
(2^8,26,52)	1.4 ms	0.55 ms	61%
(2^8,28,56)	1.5 ms	0.59 ms	60 %

Conclusion

What we have done

- „Security proof" of the general construction
- Proposal of the new scheme 0/1 UOV
  - Reduction of the public key size of UOV by 89 %
  - Speedup of the verification process by 61%
  - Known attacks cannot use the special structure of our public keys

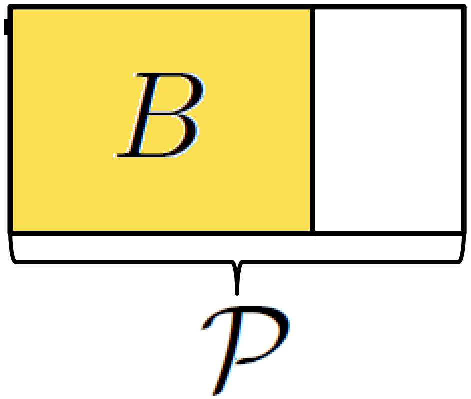


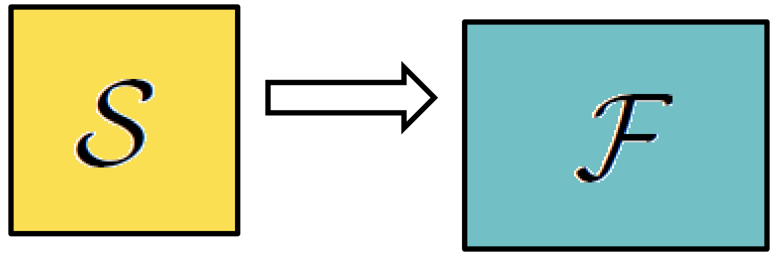
Future work

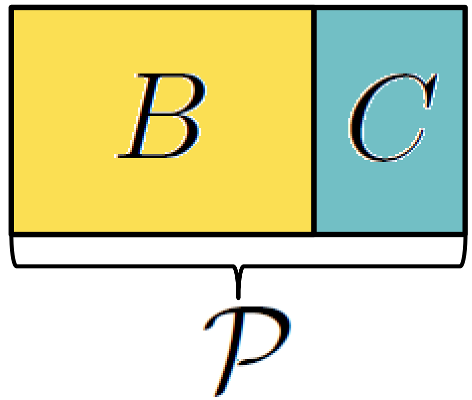
- Use of special processor instructions
- Implementation on hardware (GPU, FPGA)

Thank you for your attention

TECHNISCHE UNIVERSITÄT DARMSTADT







Questions ?