Multivariate Cryptography

2WC12 Cryptography I – Fall 2013

Ruben Niederhagen

Technische Universiteit
Eindhoven
University of Technology

December 5th, 2013

Overview

Technische Universiteit Eindhoven University of Technology

Introduction — Cryptography

Use computations that are easy (polynomial time) for the legitimate user but hard (exponential time) for an attacker.

$\Rightarrow$ Use secret knowledge (key) that makes computations easy.

Introduction — Cryptography

Use computations that are easy (polynomial time) for the legitimate user but hard (exponential time) for an attacker.

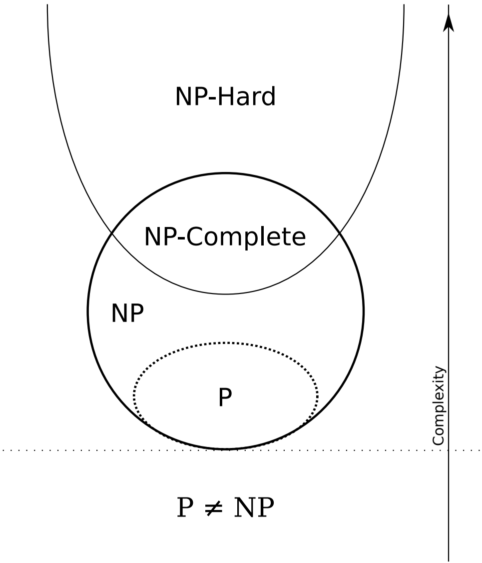
$\Rightarrow$ Use secret knowledge (key) that makes computations easy.

Commonly used hard problems:

- discrete logarithm (DLP),
- factorization,
- codes,
- lattices,
- multivariate polynomial systems,
- ...

Technische Universiteit Eindhoven University of Technology

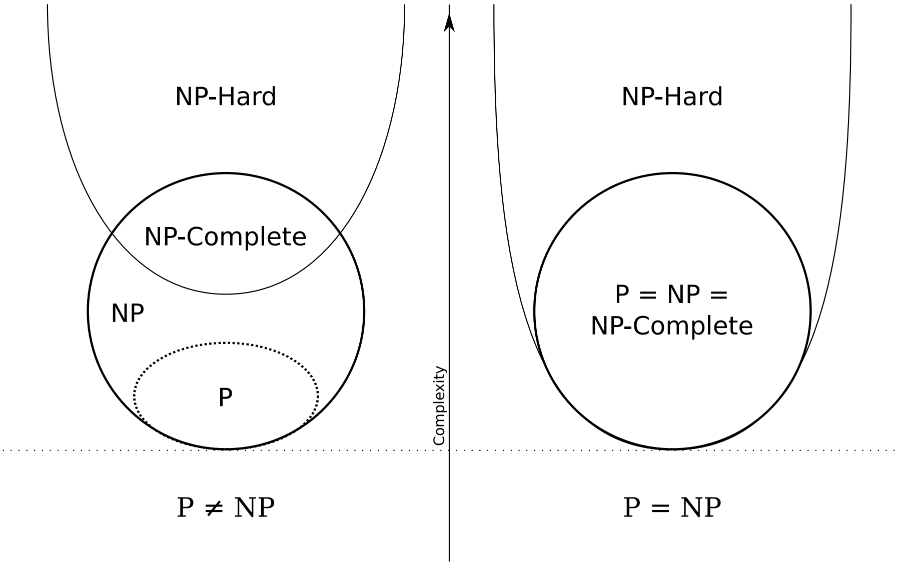
Introduction — NP-hard, NP-complete



$$P \neq NP$$

Technische Universiteit Eindhoven University of Technology

Introduction — NP-hard, NP-complete



$P \neq NP$ $\qquad$ $P = NP$

Introduction — Post-Quantum Crypto

Threat of quantum computers:

Shor's algorithm makes polynomial time:

- integer factorization
- DLP in finite fields
- DLP on elliptic curves
- DLP in general class groups

Introduction — Post-Quantum Crypto

Threat of quantum computers:

Shor's algorithm makes polynomial time:

- integer factorization
- DLP in finite fields
- DLP on elliptic curves
- DLP in general class groups

Grover's algorithm brings faster simultaneous search in data

- some security loss in symmetric crypto
  (block and stream ciphers)
- some security loss in hash functions

Compensate for Grover by doubling key size.

Technische Universiteit Eindhoven University of Technology

Introduction — Post-Quantum Crypto

The "survivors":

Public-key encryption:

- Lattice-based cryptography (e.g. NTRU, (Ring)-LWE)
- Code-based cryptography (e.g. McEliece, Niederreiter)

Public-key signatures:

- Multivariate-quadratic-equations cryptography (e.g. UOV)
- Hash based cryptography (e.g. Merkle's hash-trees signatures)

For these systems no efficient usage of Shor's algorithm is known. Grover's algorithm has to be taken into account when choosing key sizes.

Introduction — Post-Quantum Crypto

The "survivors":

Public-key encryption:

- Lattice-based cryptography (e.g. NTRU, (Ring)-LWE)
- Code-based cryptography (e.g. McEliece, Niederreiter)

Public-key signatures:

- Multivariate-quadratic-equations cryptography (e.g. UOV)
- Hash based cryptography (e.g. Merkle's hash-trees signatures)

For these systems no efficient usage of Shor's algorithm is known.
Grover's algorithm has to be taken into account when choosing key sizes.

Technische Universiteit Eindhoven University of Technology

Introduction — Multivariate Cryptography

Underlying problem:

Solving a system of m multivariate polynomial equations in n variables over $\mathbb{F}_q$ is called the MP problem.

Underlying problem:

Solving a system of m multivariate polynomial equations in n variables over $\mathbb{F}_q$ is called the MP problem.

Example:

$$5x_1^3x_2x_3^2 + 17x_2^4x_3 + 23x_1^2x_2^4 + 13x_1 + 12x_2 + 5 = 0$$
$$12x_1^2x_2^3x_3 + 15x_1x_3^3 + 25x_2x_3^3 + 5x_1 + 6x_3 + 12 = 0$$
$$28x_1x_2x_3^4 + 14x_2^3x_3^2 + 16x_1x_3 + 32x_2 + 7x_3 + 10 = 0$$

Introduction — Multivariate Cryptography

Underlying problem:

Solving a system of m multivariate polynomial equations in n variables over $\mathbb{F}_q$ is called the MP problem.

Example:

$$5x_1^3x_2x_3^2 + 17x_2^4x_3 + 23x_1^2x_2^4 + 13x_1 + 12x_2 + 5 = 0$$
$$12x_1^2x_2^3x_3 + 15x_1x_3^3 + 25x_2x_3^3 + 5x_1 + 6x_3 + 12 = 0$$
$$28x_1x_2x_3^4 + 14x_2^3x_3^2 + 16x_1x_3 + 32x_2 + 7x_3 + 10 = 0$$

Hardness:

The MP problem is an NP-complete problem even for multivariate *quadratic* systems and $q = 2$.

Underlying problem:

Solving a system of m multivariate polynomial equations in n variables over $\mathbb{F}_q$ is called the MP problem.

Example:

$$\begin{aligned} x_3x_2 + x_2x_1 + x_2 + x_1 + 1 &= 0 \\ x_3x_1 + x_3x_2 + x_3 + x_1 &= 0 \\ x_3x_2 + x_3x_1 + x_3 + x_2 &= 0 \end{aligned}$$

Hardness:

The MP problem is an NP-complete problem even for multivariate *quadratic* systems and $q = 2$.

Notation:

For a set $f = (f_1, \ldots, f_m)$ of m quadratic polynomials in n variables over $\mathbb{F}_2$, let $f(x) = (f_1(x), \ldots, f_m(x)) \in \mathbb{F}_2^m$ be the solution vector of the evaluation of f for a vector $x \in \mathbb{F}_2^n$.

TU/e Technische Universiteit Eindhoven University of Technology

Notation:

For a set $f = (f_1, \ldots, f_m)$ of m quadratic polynomials in n variables over $\mathbb{F}_2$, let $f(x) = (f_1(x), \ldots, f_m(x)) \in \mathbb{F}_2^m$ be the solution vector of the evaluation of f for a vector $x \in \mathbb{F}_2^n$.

Definition ($\mathcal{MQ}$ over $\mathbb{F}_2$):

Let $\mathcal{MQ}(\mathbb{F}_2^n, \mathbb{F}_2^m)$ be the set of all systems of quadratic equations in n variables and m equations over $\mathbb{F}_2$.
We call one element $P \in \mathcal{MQ}(\mathbb{F}_2^n, \mathbb{F}_2^m)$ an instance of $\mathcal{MQ}$ over $\mathbb{F}_2$.

Technische Universiteit Eindhoven University of Technology

Solvable in NP-time:

The following non-deterministic polynomial-time algorithm solves $\mathcal{MQ}$-$\mathbb{F}_2$ for a given system of equations:

1. Guess an assignment A for $(x_0, \ldots, x_{n-1}) \in \{0, 1\}^n$.
2. Check if all m equations are satisfied by A.
3. Output A or go to an infinity loop, respectively.

NP-hardness:

Reduce 3-SAT to $\mathcal{MQ}$-$\mathbb{F}_2$.

$$(b_1 \vee \neg b_2 \vee b_3) \wedge (b_1 \vee b_2) \wedge (\neg b_4)$$

Technische Universiteit Eindhoven University of Technology

Introduction — NP-Completeness of $\mathcal{MQ}$

NP-hardness:

Reduce 3-SAT to $\mathcal{MQ}$-$\mathbb{F}_2$.

$$(b_1 \vee \neg b_2 \vee b_3) \wedge (b_1 \vee b_2) \wedge (\neg b_4)$$

Replace all $(l_i \vee l_j)$ by $(l_i + l_j + l_i l_j)$, replace all $(l_i \vee l_j \vee l_k)$ by $(l_i + l_j + l_k + l_i l_j + l_i l_k + l_j l_k + l_i l_j l_k)$:

$$(b_1 + \neg b_2 + b_3 + b_1 \neg b_2 + b_1 b_3 + \neg b_2 b_3 + b_1 \neg b_2 b_3) \wedge (b_1 + b_2 + b_1 b_2) \wedge (\neg b_4)$$

Technische Universiteit Eindhoven University of Technology

NP-hardness:

Reduce 3-SAT to $\mathcal{MQ}$-$\mathbb{F}_2$.

$$(b_1 \vee \neg b_2 \vee b_3) \wedge (b_1 \vee b_2) \wedge (\neg b_4)$$

Replace all b_i by x_i and all $\neg b_i$ by $(1 - x_i)$:

$$\Big(x_1 + (1 - x_2) + x_3 + x_1(1 - x_2) + x_1 x_3 + (1 - x_2)x_3 + x_1(1 - x_2)x_3\Big) \wedge$$
$$(x_1 + x_2 + x_1 x_2) \wedge (1 - x_4)$$

Technische Universiteit Eindhoven University of Technology

Introduction — NP-Completeness of $\mathcal{MQ}$

NP-hardness:

Reduce 3-SAT to $\mathcal{MQ}$-$\mathbb{F}_2$.

$$(b_1 \vee \neg b_2 \vee b_3) \wedge (b_1 \vee b_2) \wedge (\neg b_4)$$

Construct an equation $e_i : c_i = 1$ for each clause c_i:

$$x_1 + (1 - x_2) + x_3 + x_1(1 - x_2) + x_1x_3 + (1 - x_2)x_3 + x_1(1 - x_2)x_3 = 1$$
$$x_1 + x_2 + x_1x_2 = 1$$
$$1 - x_4 = 1$$

Technische Universiteit Eindhoven University of Technology

NP-hardness:

Reduce 3-SAT to $\mathcal{MQ}$-$\mathbb{F}_2$.

$$(b_1 \vee \neg b_2 \vee b_3) \wedge (b_1 \vee b_2) \wedge (\neg b_4)$$

Expand all terms:

$$\begin{aligned} x_1x_2 + x_1x_2x_3 + x_2x_3 + x_2 &= 0 \\ x_1x_2 + x_1 + x_2 + 1 &= 0 \\ x_4 &= 0 \end{aligned}$$

TU/e Technische Universiteit Eindhoven University of Technology

NP-hardness:

Reduce 3-SAT to $\mathcal{MQ}$-$\mathbb{F}_2$.

$$(b_1 \vee \neg b_2 \vee b_3) \wedge (b_1 \vee b_2) \wedge (\neg b_4)$$

Iteratively add a new equation for each remaining cubic term:

$$\begin{aligned} x_1x_2 + x_5x_3 + x_2x_3 + x_2 &= 0 \\ x_1x_2 + x_1 + x_2 + 1 &= 0 \\ x_4 &= 0 \\ x_5 &= x_1x_2 \end{aligned}$$

NP-hardness:

Reduce 3-SAT to $\mathcal{MQ}$-$\mathbb{F}_2$.

$$(b_1 \vee \neg b_2 \vee b_3) \wedge (b_1 \vee b_2) \wedge (\neg b_4)$$

Final equation system:

$$\begin{aligned} x_3x_5 + x_2x_3 + x_2 + x_5 &= 0 \\ x_1 + x_2 + x_5 + 1 &= 0 \\ x_4 &= 0 \\ x_1x_2 + x_5 &= 0 \end{aligned}$$

NP-hardness:

Reduce 3-SAT to $\mathcal{MQ}$-$\mathbb{F}_2$.

$$(b_1 \vee \neg b_2 \vee b_3) \wedge (b_1 \vee b_2) \wedge (\neg b_4)$$

Final equation system:

$$\begin{aligned} x_3x_5 + x_2x_3 + x_2 + x_5 &= 0 \\ x_1 + x_2 + x_5 + 1 &= 0 \\ x_4 &= 0 \\ x_1x_2 + x_5 &= 0 \end{aligned}$$

$$\text{3-SAT} \leqslant_{\text{poly}} \mathcal{MQ}\text{-}\mathbb{F}_2$$

Introduction — NP-Completeness of $\mathcal{MQ}$

Theorem:

$\mathcal{MQ}$-$\mathbb{F}_2$ is NP-complete.

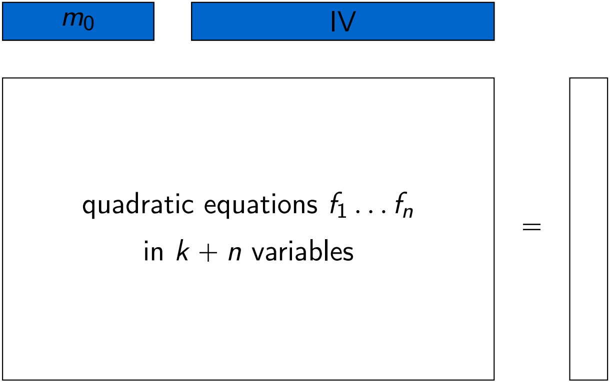
Proof.

We showed that $\mathcal{MQ}$-$\mathbb{F}_2 \in$ NP and 3-SAT $\leqslant_{\text{poly}}$ $\mathcal{MQ}$-$\mathbb{F}_2$.
Thus, $\mathcal{MQ}$-$\mathbb{F}_2$ is NP-complete. □

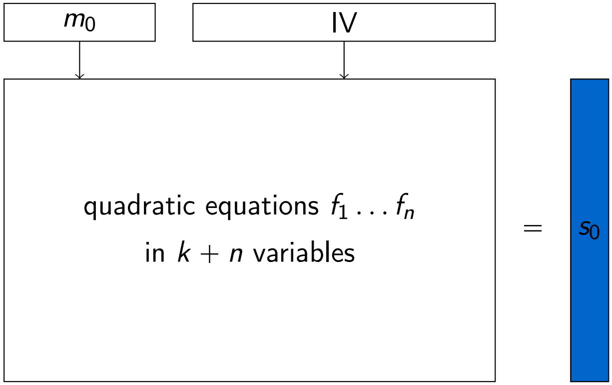
TU/e Technische Universiteit Eindhoven University of Technology

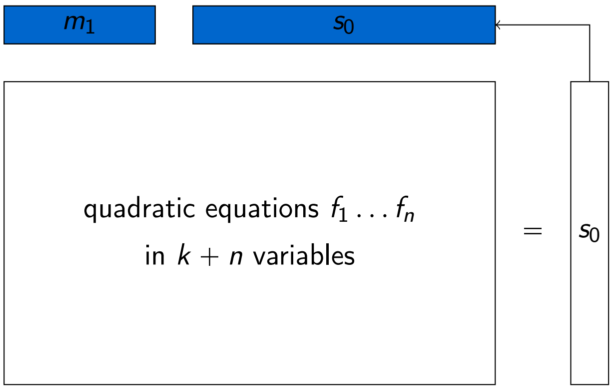
Cryptographic hash function:

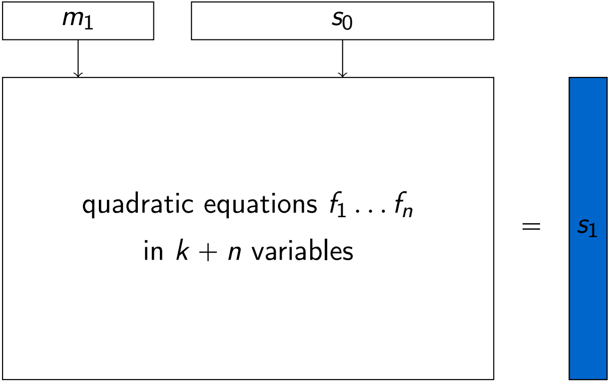
- Pre-image resistance:
  Given a hash h it should be difficult to find any message m such that $h = \mathrm{H}(m)$.
- Second pre-image resistance:
  Given an input m_0 it should be difficult to find another input m_1 such that $m_0 \neq m_1$ and $\mathrm{H}(m_0) = \mathrm{H}(m_1)$.
- Collision resistance:
  It should be difficult to find two different messages m_0 and m_1 such that that $m_0 \neq m_1$ and $\mathrm{H}(m_0) = \mathrm{H}(m_1)$.


m_0
IV
quadratic equations $f_1 \ldots f_n$
in $k + n$ variables
$=$

Cryptosystems — Hashing

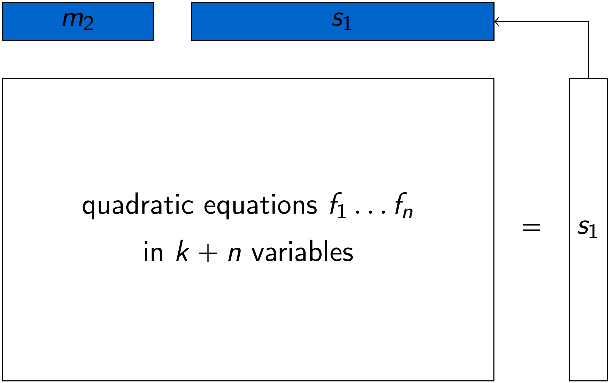


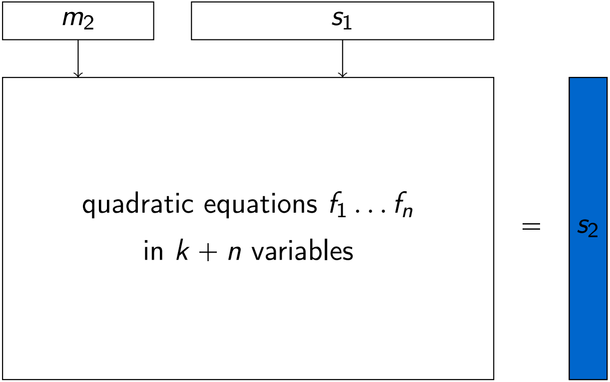

m_1
s_0
quadratic equations $f_1 \dots f_n$
in $k + n$ variables
$=$
s_0



m_1
s_0
quadratic equations $f_1 \ldots f_n$
in $k + n$ variables
$=$
s_1

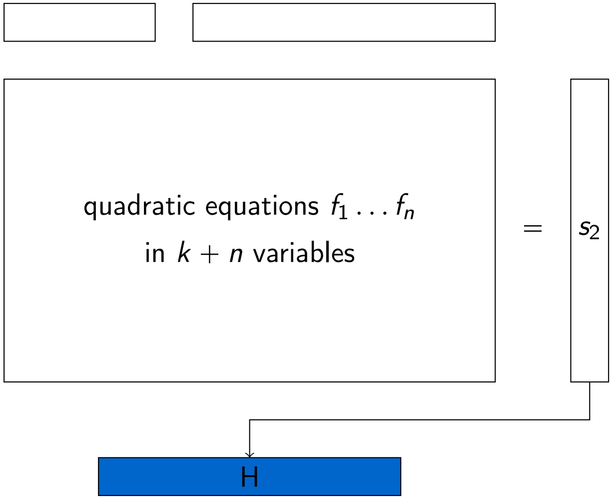
Cryptosystems — Hashing





m_2
s_1
quadratic equations $f_1 \ldots f_n$
in $k + n$ variables
$=$
s_2

Cryptosystems — Hashing



TU/e Technische Universiteit Eindhoven University of Technology

Cryptosystems — Hashing

Problem: Easy to find collisions!

$$f(m, \mathrm{IV}) = f(m', \mathrm{IV}')$$
$$f(m, \mathrm{IV}) = f(m + a, \mathrm{IV} + b)$$
$$f(m, \mathrm{IV}) - f(m + a, \mathrm{IV} + b) = 0$$

Problem: Easy to find collisions!

$$f(m, \mathrm{IV}) = f(m', \mathrm{IV}')$$
$$f(m, \mathrm{IV}) = f(m + a, \mathrm{IV} + b)$$
$$f(m, \mathrm{IV}) - f(m + a, \mathrm{IV} + b) = 0$$

$$f_0(x) = c_{2,1}x_2x_1 + c_{2,0}x_2x_0 + c_{1,0}x_1x_0 + c_2x_2 + c_1x_1 + c_0x_0 + c$$

Technische Universiteit Eindhoven University of Technology

Problem: Easy to find collisions!

$$\begin{aligned} \mathrm{f}(m, \mathrm{IV}) &= \mathrm{f}(m', \mathrm{IV}') \\ \mathrm{f}(m, \mathrm{IV}) &= \mathrm{f}(m + a, \mathrm{IV} + b) \\ \mathrm{f}(m, \mathrm{IV}) - \mathrm{f}(m + a, \mathrm{IV} + b) &= 0 \end{aligned}$$

$$f_0(x) = c_{2,1}x_2x_1 + c_{2,0}x_2x_0 + c_{1,0}x_1x_0 + c_2x_2 + c_1x_1 + c_0x_0 + c$$

$$f_0(x) - f_0(x + a) = c_{2,1}x_2x_1 + c_{2,0}x_2x_0 + c_{1,0}x_1x_0 + c_2x_2 + c_1x_1 + c_0x_0 + c$$

$$-\big(c_{2,1}(x_2 + a_2)(x_1 + a_1) + \ldots c_2(x_2 + a_2) + \cdots + c\big)$$

Cryptosystems — Hashing

Problem: Easy to find collisions!

$$\mathrm{f}(m, \mathrm{IV}) = \mathrm{f}(m', \mathrm{IV}')$$
$$\mathrm{f}(m, \mathrm{IV}) = \mathrm{f}(m + a, \mathrm{IV} + b)$$
$$\mathrm{f}(m, \mathrm{IV}) - \mathrm{f}(m + a, \mathrm{IV} + b) = 0$$

$$f_0(x) = c_{2,1}x_2x_1 + c_{2,0}x_2x_0 + c_{1,0}x_1x_0 + c_2x_2 + c_1x_1 + c_0x_0 + c$$
$$f_0(x) - f_0(x + a) = c_{2,1}x_2x_1 + c_{2,0}x_2x_0 + c_{1,0}x_1x_0 + c_2x_2 + c_1x_1 + c_0x_0 + c$$
$$-\big(c_{2,1}(x_2 + a_2)(x_1 + a_1) + \ldots c_2(x_2 + a_2) + \cdots + c\big)$$
$$f_0(x) - f_0(x + a) = \color{red}{c_{2,1}x_2x_1} + c_{2,0}x_2x_0 + c_{1,0}x_1x_0 + c_2x_2 + c_1x_1 + c_0x_0 + c$$
$$-\big(\color{red}{c_{2,1}}(\color{red}{x_2x_1} + a_1x_2 + a_2x_1 + a_1a_2) + \ldots c_2x_2 + c_2a_2 + \cdots + c\big)$$

Cryptosystems — Hashing

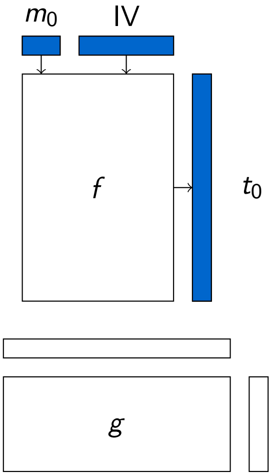
Problem: Easy to find collisions!

$$f(m, IV) = f(m', IV')$$
$$f(m, IV) = f(m + a, IV + b)$$
$$f(m, IV) - f(m + a, IV + b) = 0$$

$$f_0(x) = c_{2,1}x_2x_1 + c_{2,0}x_2x_0 + c_{1,0}x_1x_0 + c_2x_2 + c_1x_1 + c_0x_0 + c$$
$$f_0(x) - f_0(x + a) = c_{2,1}x_2x_1 + c_{2,0}x_2x_0 + c_{1,0}x_1x_0 + c_2x_2 + c_1x_1 + c_0x_0 + c$$
$$-\big(c_{2,1}(x_2 + a_2)(x_1 + a_1) + \ldots c_2(x_2 + a_2) + \cdots + c\big)$$
$$f_0(x) - f_0(x + a) = c_{2,1}x_2x_1 + c_{2,0}x_2x_0 + c_{1,0}x_1x_0 + c_2x_2 + c_1x_1 + c_0x_0 + c$$
$$-\big(c_{2,1}(x_2x_1 + a_1x_2 + a_2x_1 + a_1a_2) + \ldots c_2x_2 + c_2a_2 + \cdots + c\big)$$

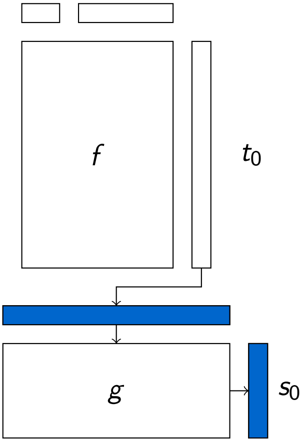
$\Rightarrow$ Underdefined linear system of $k + n$ variables and n equations!

Cryptosystems — Hashing

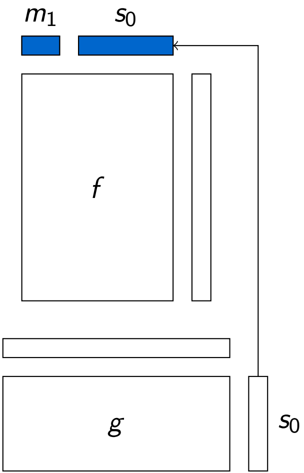


Technische Universiteit Eindhoven University of Technology

Cryptosystems — Hashing



Cryptosystems — Hashing



Example (MQ-HASH):

$f : \mathbb{F}_2^{n+k} \rightarrow \mathbb{F}_2^r$
$g : \mathbb{F}_2^r \rightarrow \mathbb{F}_2^n$

$H : (g \circ f)(s_1, \ldots, s_n, m_1, \ldots, m_k)$

MQ-HASH: $k = 32$, $n = 160$ and $r = 464$.

Composition of functions with known inverse:

Secretly choose f, g, h with known inverse functions f^{-1}, g^{-1}, h^{-1}.
Release $F = f \circ g \circ h$ as public key and h^{-1}, g^{-1}, f^{-1} as private key.

Composition of functions with known inverse:

Secretly choose f, g, h with known inverse functions f^{-1}, g^{-1}, h^{-1}.
Release $F = f \circ g \circ h$ as public key and h^{-1}, g^{-1}, f^{-1} as private key.

Example:
Choose $f = (f_1, \ldots, f_n), h = (h_1, \ldots, h_n)$ as sets of independent linear equations and

$$g(g_1, \ldots, g_n) = \begin{pmatrix} g_1 : & x_1, \\ g_2 : & x_2 + p_2(x_1), \\ g_3 : & x_3 + p_3(x_1, x_2), \\ & \ldots \\ g_4 : & x_n + p_4(x_1, \ldots, x_{n-1}) \end{pmatrix},$$

with p_i quadratic in $x_1, \ldots, x_i$.

Technische Universiteit Eindhoven University of Technology

Example:

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix},\ g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix},\ h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

Example:

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix},\ g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix},\ h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$F = f \circ g \circ h = \begin{pmatrix} x_3 x_2 + x_3 x_1 + x_2 x_1 + x_4 + x_3 + x_2 + x_1 \\ x_4 x_3 + x_4 x_1 + x_3 x_2 + x_2 x_1 + x_4 + 1 \\ x_4 x_3 + x_4 x_1 + x_3 x_1 + x_2 + 1 \\ x_3 x_2 + x_3 x_1 + x_2 x_1 + x_4 + x_1 \end{pmatrix}$$

Example (Encryption):

$$F = \begin{pmatrix} x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 \\ x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 + 1 \\ x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 \\ x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 \end{pmatrix}$$

$$F(1,0,0,1) = \begin{pmatrix} 0\cdot 0 + 0\cdot 1 + 0\cdot 1 + 1 + 0 + 0 + 1 \\ 1\cdot 0 + 1\cdot 1 + 0\cdot 0 + 0\cdot 1 + 1 + 1 \\ 1\cdot 0 + 1\cdot 1 + 0\cdot 1 + 0 + 1 \\ 0\cdot 0 + 0\cdot 1 + 0\cdot 1 + 1 + 1 \end{pmatrix} = (0,1,0,0)$$

TU/e Technische Universiteit Eindhoven University of Technology

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$f^{-1} = \begin{pmatrix} y_4 + y_3 + y_2 \\ y_3 + y_2 + y_1 + 1 \\ y_4 + y_3 + y_2 + y_1 + 1 \\ y_3 + y_1 + 1 \end{pmatrix}$$

$$f^{-1}(0, 1, 0, 0) = (1, 0, 0, 1)$$

Technische Universiteit Eindhoven University of Technology

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Technische Universiteit Eindhoven University of Technology

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Technische Universiteit Eindhoven University of Technology

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 + (\color{red}{x_1} + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Cryptosystems — Asymmetric Schemes

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 + ({\color{red}1} + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Technische Universiteit Eindhoven University of Technology

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 + (0 \cdot 1 + 0) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Technische Universiteit Eindhoven University of Technology

Cryptosystems — Asymmetric Schemes

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 + ({\color{red} x_3 x_1 + x_3 x_2 + x_1}) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 + (0 \cdot 1 + 0 \cdot 0 + 1) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 + 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Technische Universiteit Eindhoven University of Technology

Cryptosystems — Asymmetric Schemes

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$g^{-1}(1,0,0,1) = (1,0,0,0)$$

Example (Decryption):

$$f = \begin{pmatrix} x_3 + x_1 + 1 \\ x_4 + x_2 \\ x_4 + x_3 + x_1 \\ x_3 + x_2 \end{pmatrix}, g = \begin{pmatrix} x_1 \\ x_2 + (x_1 + 1) \\ x_3 + (x_2 x_1 + x_2) \\ x_4 + (x_3 x_1 + x_3 x_2 + x_1) \end{pmatrix}, h = \begin{pmatrix} x_2 + x_1 \\ x_3 + x_2 \\ x_4 + x_2 + 1 \\ x_4 + x_2 + x_1 \end{pmatrix}.$$

$$h^{-1} = \begin{pmatrix} y_4 + y_3 + 1 \\ y_4 + y_3 + y_1 + 1 \\ y_4 + y_2 + y_3 + y_1 + 1 \\ y_4 + y_1 \end{pmatrix}$$

$$h^{-1}(1,0,0,0) = (1,0,0,1)$$

Attention!

$$g(g_1, \ldots, g_n) = \begin{pmatrix} g_1: & x_1, \\ g_2: & x_2 + p_2(x_1), \\ g_3: & x_3 + p_3(x_1, x_2), \\ & \ldots \\ g_4: & x_n + p_4(x_1, \ldots, x_{n-1}) \end{pmatrix}$$

Attention!

$$g(g_1, \ldots, g_n) = \begin{pmatrix} g_1 : & x_1, \\ g_2 : & x_2 + p_2(x_1), \\ g_3 : & x_3 + p_3(x_1, x_2), \\ & \ldots \\ g_4 : & x_n + p_4(x_1, \ldots, x_{n-1}) \end{pmatrix}$$

$f \circ g \circ h$ is **<u>not</u>** a hard instance of $\mathcal{MQ}$-$\mathbb{F}_2$ due to the linearity of g_1 and g_2!

Attention!

$$g(g_1, \dots, g_n) = \begin{pmatrix} g_1: & x_1, \\ g_2: & x_2 + p_2(x_1), \\ g_3: & x_3 + p_3(x_1, x_2), \\ & \dots \\ g_4: & x_n + p_4(x_1, \dots, x_{n-1}) \end{pmatrix}$$

$f \circ g \circ h$ is **not** a hard instance of $\mathcal{MQ}$-$\mathbb{F}_2$
due to the linearity of g_1 and g_2!

Solution:

Make composition more complicated; this is ongoing research.

Cryptosystems — Asymmetric Schemes

Attention!

$$g(g_1, \ldots, g_n) = \begin{pmatrix} g_1 : & x_1, \\ g_2 : & x_2 + p_2(x_1), \\ g_3 : & x_3 + p_3(x_1, x_2), \\ & \ldots \\ g_4 : & x_n + p_4(x_1, \ldots, x_{n-1}) \end{pmatrix}$$

$f \circ g \circ h$ is **not** a hard instance of $\mathcal{MQ}$-$\mathbb{F}_2$ due to the linearity of g_1 and g_2!

Solution:

Make composition more complicated; this is ongoing research.

All asymmetric $\mathcal{MQ}$-$\mathbb{F}_2$ schemes that have been prosed so fare have been broken!

Cryptosystems — Signatures

Basic scheme:

- Signing: Encrypt message hash with private key.
- Verification: Decrypt signature with public key and compare to message hash.

Cryptosystems — Signatures

Basic scheme:

- Signing: Encrypt message hash with private key.
- Verification: Decrypt signature with public key and compare to message hash.

No secure multivariate public key system $\rightarrow$ no secure signature scheme...

Technische Universiteit Eindhoven University of Technology

Basic scheme:

- Signing: Encrypt message hash with private key.
- Verification: Decrypt signature with public key and compare to message hash.

No secure multivariate public key system $\rightarrow$ no secure signature scheme...

Wrong!

There actually are secure multivariate signature schemes that are not based on public key encryption.

TU/e Technische Universiteit Eindhoven University of Technology

Example (Oil and Vinegar):

Private key:

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, g = \begin{pmatrix} x_6x_1 + x_5x_2 + x_4x_2 + x_2x_1 + x_4 + x_3 \\ x_4x_1 + x_3x_2 + x_4 + x_1 + 1 \\ x_6x_3 + x_5x_3 + x_3x_2 + x_6 + x_5 + x_1 + 1 \end{pmatrix}$$

Cryptosystems — Signatures

Example (Oil and Vinegar):

Private key:

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, g = \begin{pmatrix} x_6x_1 + x_5x_2 + x_4x_2 + x_2x_1 + x_4 + x_3 \\ x_4x_1 + x_3x_2 + x_4 + x_1 + 1 \\ x_6x_3 + x_5x_3 + x_3x_2 + x_6 + x_5 + x_1 + 1 \end{pmatrix}$$

Public key: $g \circ f =$

$$\begin{pmatrix} x_6x_5 + x_6x_4 + x_6x_3 + x_5x_3 + x_4x_3 + x_4x_1 + x_3x_1 + x_4 + x_2 \\ x_6x_5 + x_6x_4 + x_6x_3 + x_6x_2 + x_5x_3 + x_5x_1 + x_4x_3 + x_3x_2 + x_3x_1 + x_6 + x_1 \\ x_6x_5 + x_6x_3 + x_5x_3 + x_5x_2 + x_3x_2 + x_3 + x_1 \end{pmatrix}$$

Technische Universiteit Eindhoven University of Technology

Cryptosystems — Signatures

Example (Oil and Vinegar):

Private key:

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, g = \begin{pmatrix} x_6x_1 + x_5x_2 + x_4x_2 + x_2x_1 + x_4 + x_3 \\ x_4x_1 + x_3x_2 + x_4 + x_1 + 1 \\ x_6x_3 + x_5x_3 + x_3x_2 + x_6 + x_5 + x_1 + 1 \end{pmatrix}$$

Public key: $g \circ f =$

$$\begin{pmatrix} x_6x_5 + x_6x_4 + x_6x_3 + x_5x_3 + x_4x_3 + x_4x_1 + x_3x_1 + x_4 + x_2 \\ x_6x_5 + x_6x_4 + x_6x_3 + x_6x_2 + x_5x_3 + x_5x_1 + x_4x_3 + x_3x_2 + x_3x_1 + x_6 + x_1 \\ x_6x_5 + x_6x_3 + x_5x_3 + x_5x_2 + x_3x_2 + x_3 + x_1 \end{pmatrix}$$

- Sign hash h: $s = f^{-1} \circ g^{-1}(h)$.
- Verify s: $h' = g \circ f(s)$; $h' = h$?

Technische Universiteit Eindhoven University of Technology

Example (Signing):

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, g = \begin{pmatrix} x_6x_1 + x_5x_2 + x_4x_2 + x_2x_1 + x_4 + x_3 \\ x_4x_1 + x_3x_2 + x_4 + x_1 + 1 \\ x_6x_3 + x_5x_3 + x_3x_2 + x_6 + x_5 + x_1 + 1 \end{pmatrix}$$

Cryptosystems — Signatures

Example (Signing):

Oil variables: x_6, x_5, x_4; Vinegar variables: x_3, x_2, x_1.

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, g = \begin{pmatrix} x_6x_1 + x_5x_2 + x_4x_2 + x_2x_1 + x_4 + x_3 \\ x_4x_1 + x_3x_2 + x_4 + x_1 + 1 \\ x_6x_3 + x_5x_3 + x_3x_2 + x_6 + x_5 + x_1 + 1 \end{pmatrix}$$

Technische Universiteit Eindhoven University of Technology

Example (Signing):

Oil variables: x_6, x_5, x_4; Vinegar variables: x_3, x_2, x_1.

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, g = \begin{pmatrix} x_6x_1 + x_5x_2 + x_4x_2 + x_2x_1 + x_4 + x_3 \\ x_4x_1 + x_3x_2 + x_4 + x_1 + 1 \\ x_6x_3 + x_5x_3 + x_3x_2 + x_6 + x_5 + x_1 + 1 \end{pmatrix}$$

Randomly choose x_3, x_2, x_1, e.g., $x_3 = 0, x_2 = 1, x_1 = 0$:

$$g' = \begin{pmatrix} 0x_6 + 1x_5 + 1x_4 + 1 \cdot 0 + x_4 + 0 \\ 0x_4 + 0 \cdot 1 + x_4 + 0 + 1 \\ 0x_6 + 0x_5 + 0 \cdot 1 + x_6 + x_5 + 0 + 1 \end{pmatrix}$$

Example (Signing):

Oil variables: x_6, x_5, x_4; Vinegar variables: x_3, x_2, x_1.

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, g = \begin{pmatrix} x_6x_1 + x_5x_2 + x_4x_2 + x_2x_1 + x_4 + x_3 \\ x_4x_1 + x_3x_2 + x_4 + x_1 + 1 \\ x_6x_3 + x_5x_3 + x_3x_2 + x_6 + x_5 + x_1 + 1 \end{pmatrix}$$

Randomly choose x_3, x_2, x_1, e.g., $x_3 = 0, x_2 = 1, x_1 = 0$:

$$g' = \begin{pmatrix} 0x_6 + 1x_5 + 1x_4 + 1 \cdot 0 + x_4 + 0 \\ 0x_4 + 0 \cdot 1 + x_4 + 0 + 1 \\ 0x_6 + 0x_5 + 0 \cdot 1 + x_6 + x_5 + 0 + 1 \end{pmatrix} = \begin{pmatrix} x_5 \\ x_4 + 1 \\ x_6 + x_5 + 1 \end{pmatrix}$$

Example (Signing):

Oil variables: x_6, x_5, x_4; Vinegar variables: x_3, x_2, x_1.

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, g = \begin{pmatrix} x_6x_1 + x_5x_2 + x_4x_2 + x_2x_1 + x_4 + x_3 \\ x_4x_1 + x_3x_2 + x_4 + x_1 + 1 \\ x_6x_3 + x_5x_3 + x_3x_2 + x_6 + x_5 + x_1 + 1 \end{pmatrix}$$

Sign $h = (1, 1, 0)$:

$$\begin{aligned} x_5 &= 1 \\ x_4 + 1 &= 1 \\ x_6 + x_5 + 1 &= 0 \end{aligned}$$

Cryptosystems — Signatures

Example (Signing):

Oil variables: x_6, x_5, x_4; Vinegar variables: x_3, x_2, x_1.

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, g = \begin{pmatrix} x_6x_1 + x_5x_2 + x_4x_2 + x_2x_1 + x_4 + x_3 \\ x_4x_1 + x_3x_2 + x_4 + x_1 + 1 \\ x_6x_3 + x_5x_3 + x_3x_2 + x_6 + x_5 + x_1 + 1 \end{pmatrix}$$

Sign $h = (1, 1, 0)$:

$$\begin{aligned} x_5 &= 1 \\ x_4 &= 0 \\ x_6 &= 0 \end{aligned}$$

Technische Universiteit Eindhoven University of Technology

Cryptosystems — Signatures

Example (Signing):

Oil variables: x_6, x_5, x_4; Vinegar variables: x_3, x_2, x_1.

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, g = \begin{pmatrix} x_6x_1 + x_5x_2 + x_4x_2 + x_2x_1 + x_4 + x_3 \\ x_4x_1 + x_3x_2 + x_4 + x_1 + 1 \\ x_6x_3 + x_5x_3 + x_3x_2 + x_6 + x_5 + x_1 + 1 \end{pmatrix}$$

Sign $h = (1, 1, 0)$:

$$x_5 = 1$$
$$x_4 = 0$$
$$x_6 = 0$$

$$g^{-1}(1, 1, 0) = (0, 1, 0, 0, 1, 0)$$

Example (Signing):

$g^{-1}(1,1,0) = (0,1,0,0,1,0)$

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix},$$

Cryptosystems — Signatures

Example (Signing):

$g^{-1}(1,1,0) = (0,1,0,0,1,0)$

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, f^{-1} = \begin{pmatrix} x_2 + x_1 + 1 \\ x_6 + x_5 + x_3 + x_2 + x_1 + 1 \\ x_6 + x_3 + x_2 + x_1 \\ x_6 + x_5 + x_4 + x_3 + x_2 + x_1 \\ x_6 + x_2 + x_1 + 1 \\ x_6 + x_3 + x_2 + 1 \end{pmatrix}$$

Example (Signing):

$$g^{-1}(1,1,0) = (0,1,0,0,1,0)$$

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, f^{-1} = \begin{pmatrix} x_2 + x_1 + 1 \\ x_6 + x_5 + x_3 + x_2 + x_1 + 1 \\ x_6 + x_3 + x_2 + x_1 \\ x_6 + x_5 + x_4 + x_3 + x_2 + x_1 \\ x_6 + x_2 + x_1 + 1 \\ x_6 + x_3 + x_2 + 1 \end{pmatrix}$$

$$f^{-1}(0,1,0,0,1,0) = (0,0,0,1,1,0)$$

Example (Signing):

$$g^{-1}(1,1,0) = (0,1,0,0,1,0)$$

$$f = \begin{pmatrix} x_6 + x_3 + 1 \\ x_6 + x_3 + x_1 \\ x_5 + x_3 + 1 \\ x_4 + x_2 + 1 \\ x_3 + x_2 + 1 \\ x_5 + x_1 \end{pmatrix}, f^{-1} = \begin{pmatrix} x_2 + x_1 + 1 \\ x_6 + x_5 + x_3 + x_2 + x_1 + 1 \\ x_6 + x_3 + x_2 + x_1 \\ x_6 + x_5 + x_4 + x_3 + x_2 + x_1 \\ x_6 + x_2 + x_1 + 1 \\ x_6 + x_3 + x_2 + 1 \end{pmatrix}$$

$$f^{-1}(0,1,0,0,1,0) = (0,0,0,1,1,0)$$

$$s = f^{-1}g^{-1}(1,1,0) = (0,0,0,1,1,0)$$

Example (Verification):

$h = (1,1,0), s = (0,0,0,1,1,0)$

Cryptosystems — Signatures

Example (Verification):

$h = (1,1,0), s = (0,0,0,1,1,0)$

$$g \circ f = \begin{pmatrix} x_6x_5 + x_6x_4 + x_6x_3 + x_5x_3 + x_4x_3 + x_4x_1 + x_3x_1 + x_4 + x_2 \\ x_6x_5 + x_6x_4 + x_6x_3 + x_6x_2 + x_5x_3 + x_5x_1 + x_4x_3 + x_3x_2 + x_3x_1 + x_6 + x_1 \\ x_6x_5 + x_6x_3 + x_5x_3 + x_5x_2 + x_3x_2 + x_3 + x_1 \end{pmatrix}$$

Example (Verification):

$h = (1, 1, 0), s = (0, 0, 0, 1, 1, 0)$

$$g \circ f = \begin{pmatrix} x_6x_5 + x_6x_4 + x_6x_3 + x_5x_3 + x_4x_3 + x_4x_1 + x_3x_1 + x_4 + x_2 \\ x_6x_5 + x_6x_4 + x_6x_3 + x_6x_2 + x_5x_3 + x_5x_1 + x_4x_3 + x_3x_2 + x_3x_1 + x_6 + x_1 \\ x_6x_5 + x_6x_3 + x_5x_3 + x_5x_2 + x_3x_2 + x_3 + x_1 \end{pmatrix}$$

$h' = g \circ f(0, 0, 0, 1, 1, 0) = (1, 1, 0)$

TU/e Technische Universiteit Eindhoven University of Technology

Public key encryption scheme?

Oil and Vinegar can not be used as public key encryption scheme due to the randomness of the vinegar variables.

Technische Universiteit
Eindhoven
University of Technology

Public key encryption scheme?

Oil and Vinegar can not be used as public key encryption scheme due to the randomness of the vinegar variables.

Oil and Vinegar is broken!

Technische Universiteit Eindhoven University of Technology

Public key encryption scheme?

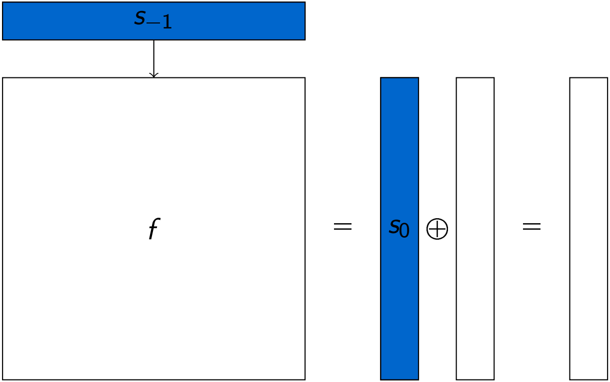
Oil and Vinegar can not be used as public key encryption scheme due to the randomness of the vinegar variables.

Oil and Vinegar is broken!

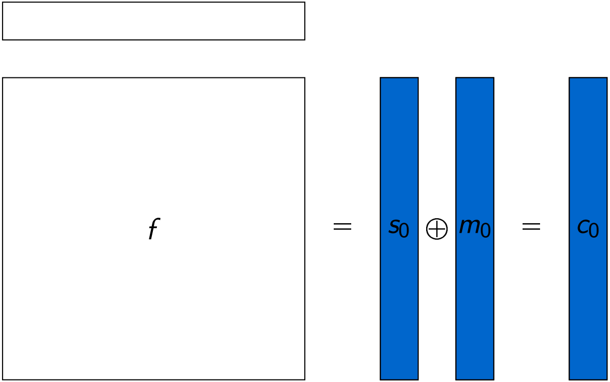
There are variations of Oil and Vinegar, e.g., Unbalanced Oil and Vinegar (UOB), that are (not yet) broken.

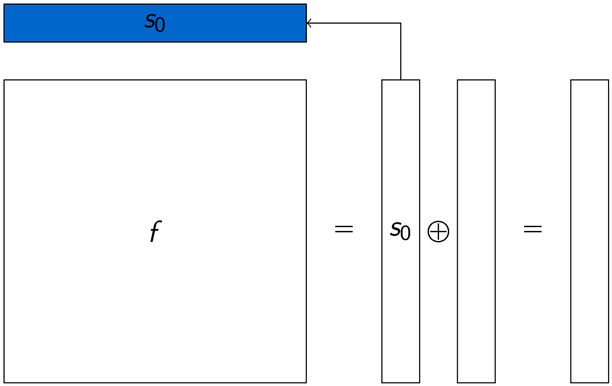
TU/e Technische Universiteit Eindhoven University of Technology

Pre-process symmetric key and IV to obtain initial state s_{-1}.



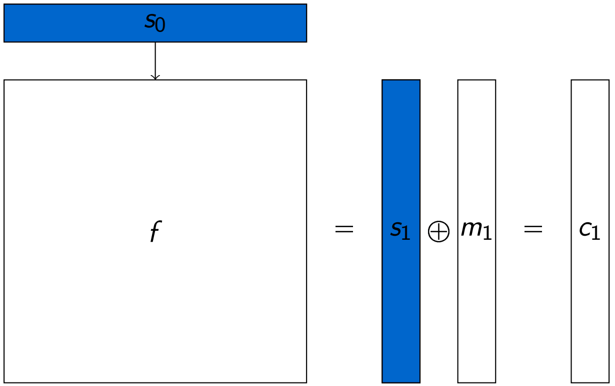
Cryptosystems — Symmetric Schemes



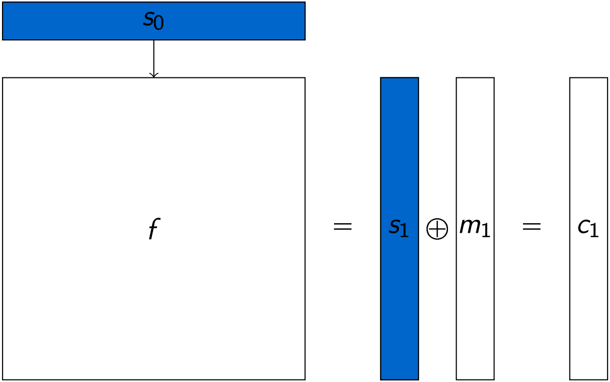


TU/e Technische Universiteit Eindhoven University of Technology

Cryptosystems — Symmetric Schemes

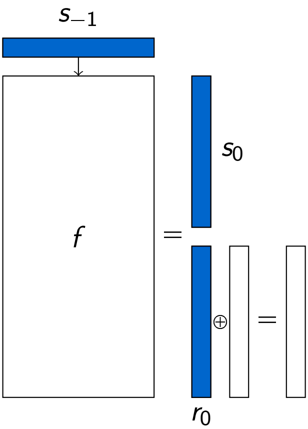


Cryptosystems — Symmetric Schemes

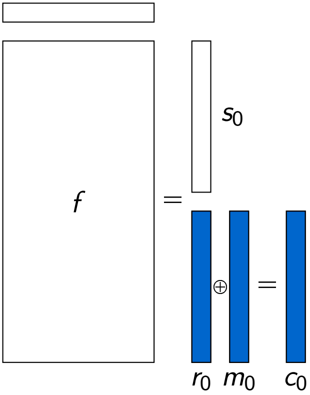


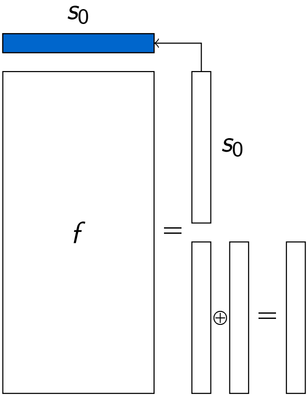
Easy to obtain key stream with a single known plain text block!

Pre-process symmetric key and IV to obtain initial state s_{-1}.



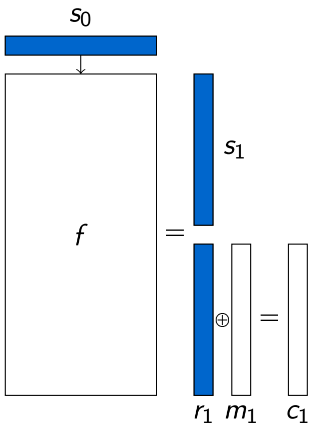
Cryptosystems — Symmetric Schemes





s_0
s_0
f
$=$
$\oplus$
$=$

Cryptosystems — Symmetric Schemes



Cryptosystems — Symmetric Schemes

QUAD stream cipher

Provable secure!

QUAD stream cipher

“Provable secure!”

Suggested parameters QUAD(256,20,20) have been broken!

QUAD stream cipher

"Provable secure!"

Suggested parameters QUAD(256,20,20) have been broken!

Parameters that are still considered secure:
QUAD(2,160,160), QUAD(2,256,256), QUAD(2,350,350), . . .

System Solving

Algebraic Cryptanalysis:

Obtain a system of multivariate polynomial equations with the secret among the variables.

- Naturally breaks multivariate crypto schemes,
- does not break AES as first advertised,
- but does break, e.g., KeeLoq.

TU/e Technische Universiteit Eindhoven University of Technology

Example:

$$F = \begin{pmatrix} x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 \\ x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 + 1 \\ x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 \\ x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 \end{pmatrix}$$

Find x for $F(x) = (0, 1, 0, 0)$.

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 + 1 = 1 \quad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \quad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \quad (4)$$

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1)$$
$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 + 1 = 1 \quad (2)$$
$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \quad (3)$$
$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \quad (4)$$

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \qquad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \qquad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \qquad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \qquad (4)$$

Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \qquad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \qquad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \qquad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \qquad (4)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_2 + 1 = 0 \qquad (2) + (3) = \qquad (5)$$

Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \qquad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \qquad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \qquad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \qquad (4)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_2 + 1 = 0 \qquad (2) + (3) = \qquad (5)$$

$$x_2 + x_1 + 1 = 0 \qquad (4) + (5) = \qquad (6)$$

Technische Universiteit Eindhoven University of Technology

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \qquad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \qquad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \qquad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \qquad (4)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_2 + 1 = 0 \qquad (2) + (3) = \qquad (5)$$

$$x_2 + x_1 + 1 = 0 \qquad (4) + (5) = \qquad (6)$$

$$x_3 + x_2 = 0 \qquad (1) + (4) = \qquad (7)$$

System Solving — Gröbner Bases

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \qquad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \qquad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \qquad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \qquad (4)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_2 + 1 = 0 \qquad (2) + (3) = \qquad (5)$$

$$x_2 + x_1 + 1 = 0 \qquad (4) + (5) = \qquad (6)$$

$$x_3 + x_2 = 0 \qquad (1) + (4) = \qquad (7)$$

$$\color{red}{x_3x_2x_1} + x_4x_1 + x_3x_2 + x_2x_1 + x_4 + x_3 = 0 \qquad x_3(1) + (2) = \qquad (8)$$

TU/e Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \qquad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \qquad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \qquad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \qquad (4)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_2 + 1 = 0 \qquad (2) + (3) = \qquad (5)$$

$$x_2 + x_1 + 1 = 0 \qquad (4) + (5) = \qquad (6)$$

$$x_3 + x_2 = 0 \qquad (1) + (4) = \qquad (7)$$

$$x_3x_2x_1 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 + x_3 = 0 \qquad x_3(1) + (2) = \qquad (8)$$

$$x_3x_2x_1 + x_4x_1 + x_3x_2 + x_3x_1 + x_2 + 1 = 0 \qquad x_3(4) + (3) = \qquad (9)$$

TU/e Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \qquad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \qquad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \qquad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \qquad (4)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_2 + 1 = 0 \qquad (2) + (3) = \qquad (5)$$

$$x_2 + x_1 + 1 = 0 \qquad (4) + (5) = \qquad (6)$$

$$x_3 + x_2 = 0 \qquad (1) + (4) = \qquad (7)$$

$$x_3x_2x_1 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 + x_3 = 0 \qquad x_3(1) + (2) = \qquad (8)$$

$$x_3x_2x_1 + x_4x_1 + x_3x_2 + x_3x_1 + x_2 + 1 = 0 \qquad x_3(4) + (3) = \qquad (9)$$

$$x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + 1 = 0 \qquad (8) + (9) = \qquad (10)$$

TU/e Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \qquad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \qquad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \qquad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \qquad (4)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_2 + 1 = 0 \qquad (2) + (3) = \qquad (5)$$

$$x_2 + x_1 + 1 = 0 \qquad (4) + (5) = \qquad (6)$$

$$x_3 + x_2 = 0 \qquad (1) + (4) = \qquad (7)$$

$$x_3x_2x_1 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 + x_3 = 0 \qquad x_3(1) + (2) = \qquad (8)$$

$$x_3x_2x_1 + x_4x_1 + x_3x_2 + x_3x_1 + x_2 + 1 = 0 \qquad x_3(4) + (3) = \qquad (9)$$

$$x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + 1 = 0 \qquad (8) + (9) = \qquad (10)$$

$$x_4 + x_3 + x_2 + 1 = 0 \qquad x_1(7) + (10) = \qquad (11)$$

Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Example:

$$
\begin{aligned}
x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 &= 0 && && (1)\\
x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 &= 0 && && (2)\\
x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 &= 0 && && (3)\\
x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 &= 0 && && (4)\\
x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_2 + 1 &= 0 && (2) + (3) = && (5)\\
x_2 + x_1 + 1 &= 0 && (4) + (5) = && (6)\\
x_3 + x_2 &= 0 && (1) + (4) = && (7)\\
x_3x_2x_1 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 + x_3 &= 0 && x_3(1) + (2) = && (8)\\
x_3x_2x_1 + x_4x_1 + x_3x_2 + x_3x_1 + x_2 + 1 &= 0 && x_3(4) + (3) = && (9)\\
x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + 1 &= 0 && (8) + (9) = && (10)\\
x_4 + x_3 + x_2 + 1 &= 0 && x_1(7) + (10) = && (11)\\
{\color{red} x_4 + 1} &= 0 && (7) + (11) = && (12)
\end{aligned}
$$

TU/e Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Example:

$$\begin{aligned}
x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 &= 0 && && (1)\\
x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 &= 0 && && (2)\\
x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 &= 0 && && (3)\\
x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 &= 0 && && (4)\\
x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_2 + 1 &= 0 && (2) + (3) = && (5)\\
x_2 + x_1 + 1 &= 0 && (4) + (5) = && (6)\\
x_3 + x_2 &= 0 && (1) + (4) = && (7)\\
x_3x_2x_1 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 + x_3 &= 0 && x_3(1) + (2) = && (8)\\
x_3x_2x_1 + x_4x_1 + x_3x_2 + x_3x_1 + x_2 + 1 &= 0 && x_3(4) + (3) = && (9)\\
x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + 1 &= 0 && (8) + (9) = && (10)\\
x_4 + x_3 + x_2 + 1 &= 0 && x_1(7) + (10) = && (11)\\
x_4 &= 1 && (7) + (11) = && (12)
\end{aligned}$$

TU/e Technische Universiteit Eindhoven University of Technology

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \quad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \quad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \quad (4)$$

$$x_2 + x_1 + 1 = 0 \quad (6)$$

$$x_3 + x_2 = 0 \quad (7)$$

$$x_4 = 1 \quad (12)$$

Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1)$$
$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \quad (2)$$
$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \quad (3)$$
$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \quad (4)$$
$$x_2 + x_1 + 1 = 0 \quad (6)$$
$$x_3 + x_2 = 0 \quad (7)$$
$$x_4 = 1 \quad (12)$$
$$x_4x_3x_1 + x_4x_3 + x_2x_1 + x_4 + x_3 + x_1 = 0 \quad x_3(3) + (4) = \quad (13)$$

Technische Universiteit Eindhoven University of Technology

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \quad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \quad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \quad (4)$$

$$x_2 + x_1 + 1 = 0 \quad (6)$$

$$x_3 + x_2 = 0 \quad (7)$$

$$x_4 = 1 \quad (12)$$

$$x_4x_3x_1 + x_4x_3 + x_2x_1 + x_4 + x_3 + x_1 = 0 \quad x_3(3) + (4) = \quad (13)$$

$$x_4x_3x_1 + x_3x_2x_1 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1) + x_3(2) = \quad (14)$$

TU/e Technische Universiteit Eindhoven University of Technology

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \quad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \quad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \quad (4)$$

$$x_2 + x_1 + 1 = 0 \quad (6)$$

$$x_3 + x_2 = 0 \quad (7)$$

$$x_4 = 1 \quad (12)$$

$$x_4x_3x_1 + x_4x_3 + x_2x_1 + x_4 + x_3 + x_1 = 0 \quad x_3(3) + (4) = \quad (13)$$

$$x_4x_3x_1 + x_3x_2x_1 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1) + x_3(2) = \quad (14)$$

$$x_2 = 0 \qquad (14) + (13) + (9) + x_4(7) + x_4(6) + x_2(7) + (12) = \qquad (15)$$

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1)$$
$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \quad (2)$$
$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \quad (3)$$
$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \quad (4)$$
$$x_2 + x_1 + 1 = 0 \quad (6)$$
$$x_3 + x_2 = 0 \quad (7)$$
$$x_4 = 1 \quad (12)$$
$$x_4x_3x_1 + x_4x_3 + x_2x_1 + x_4 + x_3 + x_1 = 0 \quad x_3(3) + (4) = \quad (13)$$
$$x_4x_3x_1 + x_3x_2x_1 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1) + x_3(2) = \quad (14)$$

$$x_2 = 0 \quad (14) + (13) + (9) + x_4(7) + x_4(6) + x_2(7) + (12) = \quad (15)$$
$$x_3 = 0 \quad (7) + (15) = \quad (16)$$

TU/e Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1)$$
$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \quad (2)$$
$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \quad (3)$$
$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \quad (4)$$
$$x_2 + x_1 + 1 = 0 \quad (6)$$
$$x_3 + x_2 = 0 \quad (7)$$
$$x_4 = 1 \quad (12)$$
$$x_4x_3x_1 + x_4x_3 + x_2x_1 + x_4 + x_3 + x_1 = 0 \quad x_3(3) + (4) = \quad (13)$$
$$x_4x_3x_1 + x_3x_2x_1 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1) + x_3(2) = \quad (14)$$

$$x_2 = 0 \quad (14) + (13) + (9) + x_4(7) + x_4(6) + x_2(7) + (12) = \quad (15)$$
$$x_3 = 0 \quad (7) + (15) = \quad (16)$$
$$x_1 = 1 \quad (6) + (15) = \quad (17)$$

Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Example:

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1)$$

$$x_4x_3 + x_4x_1 + x_3x_2 + x_2x_1 + x_4 = 0 \quad (2)$$

$$x_4x_3 + x_4x_1 + x_3x_1 + x_2 + 1 = 0 \quad (3)$$

$$x_3x_2 + x_3x_1 + x_2x_1 + x_4 + x_1 = 0 \quad (4)$$

$$x_2 + x_1 + 1 = 0 \quad (6)$$

$$x_3 + x_2 = 0 \quad (7)$$

$$x_4 = 1 \quad (12)$$

$$x_4x_3x_1 + x_4x_3 + x_2x_1 + x_4 + x_3 + x_1 = 0 \quad x_3(3) + (4) = \quad (13)$$

$$x_4x_3x_1 + x_3x_2x_1 + x_3x_1 + x_2x_1 + x_4 + x_3 + x_2 + x_1 = 0 \quad (1) + x_3(2) = \quad (14)$$

$$x_2 = 0 \quad (14) + (13) + (9) + x_4(7) + x_4(6) + x_2(7) + (12) = \quad (15)$$

$$x_3 = 0 \quad (7) + (15) = \quad (16)$$

$$x_1 = 1 \quad (6) + (15) = \quad (17)$$

Technische Universiteit Eindhoven University of Technology

System Solving — Gröbner Bases

Algorithm due to Buchberger:

- Transform set of equations to a Gröbner basis; obtain solution of the system from the final representation.
- During computation, the maximum degree increases to $D > 2$.
- There are several improvements of Buchbergers algorithm, e.g., Faugère's F_4 and F_5 (implemented, e.g., in Magma).

Technische Universiteit Eindhoven University of Technology

System Solving — Extended Linearization

The XL algorithm

- *XL* is an acronym for *extended linearization*:
  - *extend* a quadratic system by multiplying with appropriate monomials,
  - *linearize* by treating each monomial as an independent variable,
  - solve the linearized system.
- Special case of Gröbner basis algorithms.
- First suggested by Lazard (1983).
- Reinvented by Courtois, Klimov, Patarin, and Shamir (2000).
- More "easy" to parallelize compared to Gröbner basis solvers.

System Solving — Extended Linearization

Basic idea:

For $b \in \mathbb{N}^n$ denote by x^b the monomial $x_1^{b_1} x_2^{b_2} \dots x_n^{b_n}$ and by $|b| = b_1 + b_2 + \cdots + b_n$ the total degree of x^b.

given:	finite field $K = \mathbb{F}_q$
	system $\mathcal{A}$ of m multivariate quadratic equations:
	$\ell_1 = \ell_2 = \cdots = \ell_m = 0,\ \ell_i \in K[x_1, x_2, \dots, x_n]$
choose:	operational degree $D \in \mathbb{N}$
extend:	system $\mathcal{A}$ to the system
	$\mathcal{R}^{(D)} = \{x^b \ell_i = 0 : \lvert b \rvert \leqslant D - 2, \ell_i \in \mathcal{A}\}$
linearize:	consider $x^d, d \leqslant D$ a new variable, obtain linear system $\mathcal{M}$
solve:	linear system $\mathcal{M}$

Technische Universiteit Eindhoven University of Technology

System Solving — Extended Linearization

Basic idea:

For $b \in \mathbb{N}^n$ denote by x^b the monomial $x_1^{b_1} x_2^{b_2} \dots x_n^{b_n}$ and by $|b| = b_1 + b_2 + \cdots + b_n$ the total degree of x^b.

given:	finite field $K = \mathbb{F}_q$
	system $\mathcal{A}$ of m multivariate quadratic equations:
	$\ell_1 = \ell_2 = \cdots = \ell_m = 0,\ \ell_i \in K[x_1, x_2, \dots, x_n]$
choose:	<u>operational degree $D \in \mathbb{N}$</u> How?
extend:	system $\mathcal{A}$ to the system
	$\mathcal{R}^{(D)} = \{x^b \ell_i = 0 : \lvert b\rvert \leqslant D-2, \ell_i \in \mathcal{A}\}$
linearize:	consider $x^d, d \leqslant D$ a new variable, obtain linear system $\mathcal{M}$
solve:	linear system $\mathcal{M}$

TU/e Technische Universiteit Eindhoven University of Technology

System Solving — Extended Linearization

Basic idea:

For $b \in \mathbb{N}^n$ denote by x^b the monomial $x_1^{b_1} x_2^{b_2} \dots x_n^{b_n}$ and by $|b| = b_1 + b_2 + \cdots + b_n$ the total degree of x^b.

given: finite field $K = \mathbb{F}_q$
system $\mathcal{A}$ of m multivariate quadratic equations:
$\ell_1 = \ell_2 = \cdots = \ell_m = 0,\ \ell_i \in K[x_1, x_2, \dots, x_n]$

choose: operational degree $D \in \mathbb{N}$ How?

extend: system $\mathcal{A}$ to the system
$\mathcal{R}^{(D)} = \{x^b \ell_i = 0 : |b| \leqslant D - 2, \ell_i \in \mathcal{A}\}$

linearize: consider $x^d, d \leqslant D$ a new variable, obtain linear system $\mathcal{M}$

solve: linear system $\mathcal{M}$

minimum degree D_0 for reliable termination (Yang and Chen):

$$D_0 := \min\{D : ((1 - \lambda)^{m-n-1}(1 + \lambda)^m)[D] \leqslant 0\}$$

TU/e Technische Universiteit Eindhoven University of Technology

System Solving — Extended Linearization

Basic idea:

For $b \in \mathbb{N}^n$ denote by x^b the monomial $x_1^{b_1} x_2^{b_2} \dots x_n^{b_n}$ and by $|b| = b_1 + b_2 + \cdots + b_n$ the total degree of x^b.

given:	finite field $K = \mathbb{F}_q$
	system $\mathcal{A}$ of m multivariate quadratic equations:
	$\ell_1 = \ell_2 = \cdots = \ell_m = 0,\ \ell_i \in K[x_1, x_2, \dots, x_n]$
choose:	operational degree $D \in \mathbb{N}$ How?
extend:	system $\mathcal{A}$ to the system
	$\mathcal{R}^{(D)} = \{x^b \ell_i = 0 : \lvert b\rvert \leqslant D - 2, \ell_i \in \mathcal{A}\}$
linearize:	consider $x^d, d \leqslant D$ a new variable, obtain linear system $\mathcal{M}$
solve:	linear system $\mathcal{M}$ How?

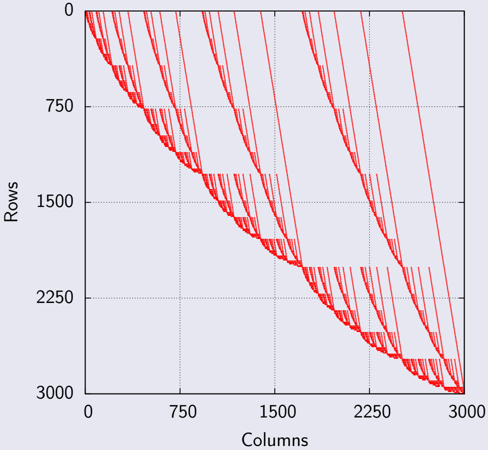
minimum degree D_0 for reliable termination (Yang and Chen):

$$D_0 := \min\{D : ((1 - \lambda)^{m-n-1}(1 + \lambda)^m)[D] \leqslant 0\}$$

TU/e Technische Universiteit Eindhoven University of Technology

System Solving — Extended Linearization

Solve the sparse linear system $\mathcal{M}$:



Use, e.g., the (block) Lanczos or the (block) Wiedemann algorithm.

System Solving — Brute Force

Efficiency:

Gröbner basis solvers and XL are efficient for solving multivariate polynomial systems over *large* finite fields.

System Solving — Brute Force

Efficiency:

Gröbner basis solvers and XL are efficient for solving multivariate polynomial systems over *large* finite fields.

Most Efficient Algorithm for $\mathbb{F}_2$:

Brute-force search, testing all 2^n possible inputs.

Exhaustive Search — Approach

Full-Evaluation Approach

- Evaluate the whole equation for each possible input.
- Time Complexity: $O(2^n n^2)$
- Memory Complexity: $O(n)$

TU/e Technische Universiteit Eindhoven University of Technology

Exhaustive Search — Approach

Full-Evaluation Approach

- Evaluate the whole equation for each possible input.
- Time Complexity: $O(2^n n^2)$
- Memory Complexity: $O(n)$

Gray-Code Approach

- Only re-compute those parts of the equation that have changed.
- Enumerate input vector in Gray-code order.
- Update solution using the derivatives of the involved variables.
- Time Complexity: $O(2^n m)$
- Memory Complexity: $O(n^2 m)$

Trade computation for memory.

Gray-Code Approach

$k = 01010_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ x_0 = 0$

$$f = x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1$$
$$f = 0 \cdot 0 + 1 \cdot 0 + 0 \cdot 1 + 1 + 1 + 0 + 1$$

Gray-Code Approach

$k = 01010_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ x_0 = 0$

$$\begin{aligned} f &= x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1 \\ f &= 0\cdot 0 + 1\cdot 0 + 0\cdot 1 + 1 + 1 + 0 + 1 \end{aligned}$$

$k = 0101{\color{red}1}_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ {\color{red}x_0 = 1}$

$$\begin{aligned} f &= x_4x_2 + x_3{\color{red}x_0} + x_2x_1 + x_3 + x_1 + {\color{red}x_0} + 1 \\ f &= 0\cdot 0 + 1\cdot {\color{red}1} + 0\cdot 1 + 1 + 1 + {\color{red}1} + 1 \end{aligned}$$

Gray-Code Approach

$k = 01010_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ x_0 = 0$

$$\begin{aligned} f &= x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1 \\ f &= 0 \cdot 0 + 1 \cdot 0 + 0 \cdot 1 + 1 + 1 + 0 + 1 \end{aligned}$$

$k = 01011_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ x_0 = 1$

$$\begin{aligned} f &= x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1 \\ f &= 0 \cdot 0 + 1 \cdot 1 + 0 \cdot 1 + 1 + 1 + 1 + 1 \end{aligned}$$

$k = 01100_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 1,\ x_1 = 0,\ x_0 = 0$

$$\begin{aligned} f &= x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1 \\ f &= 0 \cdot 1 + 1 \cdot 0 + 1 \cdot 0 + 1 + 0 + 0 + 1 \end{aligned}$$

Gray-Code Approach

$k = 01010_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ x_0 = 0$

$$\begin{array}{ll} f = & x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1 \\ f = & 0\cdot 0 + 1\cdot 0 + 0\cdot 1 + 1 + 1 + 0 + 1 \end{array}$$

$k = 01011_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ x_0 = 1$

$$\begin{array}{ll} f = & x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1 \\ f = & 0\cdot 0 + 1\cdot 1 + 0\cdot 1 + 1 + 1 + 1 + 1 \end{array}$$

$k = 01001_b$ in *Gray-code* order

$$\begin{array}{ll} f = & x_4x_2 + x_3x_0 + x_2\color{red}{x_1} + x_3 + \color{red}{x_1} + x_0 + 1 \\ f = & 0\cdot 0 + 1\cdot 1 + 0\cdot \color{red}{0} + 1 + \color{red}{0} + 1 + 1 \end{array}$$

TU/e Technische Universiteit Eindhoven University of Technology

Gray-Code Approach

$k = 01010_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ x_0 = 0$

$$\begin{aligned} f &= x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1 \\ f &= 0\cdot 0 + 1\cdot 0 + 0\cdot 1 + 1 + 1 + 0 + 1 \end{aligned}$$

$k = 01011_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ x_0 = 1$

$$\begin{aligned} f &= x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1 \\ f &= 0\cdot 0 + 1\cdot 1 + 0\cdot 1 + 1 + 1 + 1 + 1 \end{aligned}$$

$k = 01001_b$ in *Gray-code* order

$$\begin{aligned} f &= x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1 \\ f &= 0\cdot 0 + 1\cdot 1 + 0\cdot 0 + 1 + 0 + 1 + 1 \\ f &= f(01011_b) - 0\cdot 1 - 1 + 0\cdot 0 + 0 \end{aligned}$$

Technische Universiteit Eindhoven University of Technology

Gray-Code Approach

$k = 01010_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ x_0 = 0$

$$f = x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1$$
$$f = 0\cdot 0 + 1\cdot 0 + 0\cdot 1 + 1 + 1 + 0 + 1$$

$k = 01011_b;\ x_4 = 0,\ x_3 = 1,\ x_2 = 0,\ x_1 = 1,\ x_0 = 1$

$$f = x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1$$
$$f = 0\cdot 0 + 1\cdot 1 + 0\cdot 1 + 1 + 1 + 1 + 1$$

$k = 01001_b$ in *Gray-code* order

$$f = x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1$$
$$f = 0\cdot 0 + 1\cdot 1 + 0\cdot 0 + 1 + 0 + 1 + 1$$
$$f = f(01011_b) - 0\cdot 1 - 1 + 0\cdot 0 + 0$$
$$f = f(01011_b) + \frac{\partial f}{\partial x_1}(01001_b)$$

Technische Universiteit Eindhoven University of Technology

Exhaustive Search — Approach

Full-Evaluation Approach

- Evaluate the whole equation for each possible input.
- Time Complexity: $O(2^n n^2)$
- Memory Complexity: $O(n)$

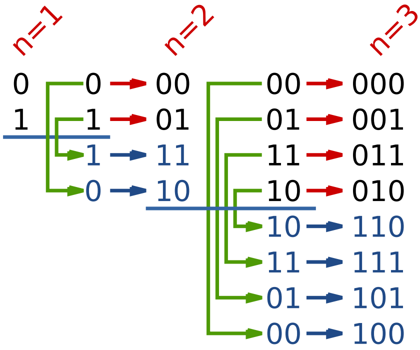
Gray-Code Approach

- Only re-compute those parts of the equation that have changed.
- Enumerate input vector in Gray-code order.
- Update solution using the derivatives of the involved variables.
- Time Complexity: $O(2^n m)$
- Memory Complexity: $O(n^2 m)$

Trade computation for memory.

TU/e Technische Universiteit Eindhoven University of Technology

Gray Code



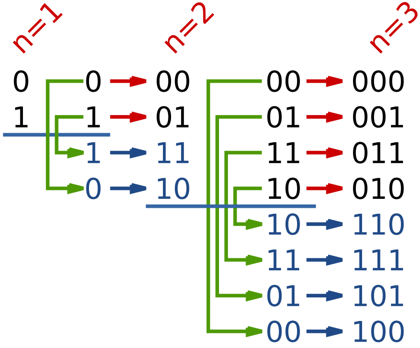
Binary to Gray:

```
(ctr >> 1) ^ ctr
```

ctr	bin	gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

TU/e Technische Universiteit Eindhoven University of Technology

Gray Code



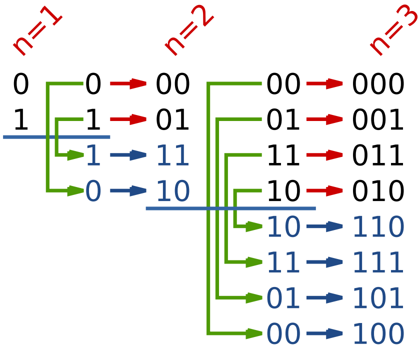
Binary to Gray:

```
(ctr >> 1) ^ ctr
```

ctr	bin	gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

TU/e Technische Universiteit Eindhoven University of Technology

Gray Code



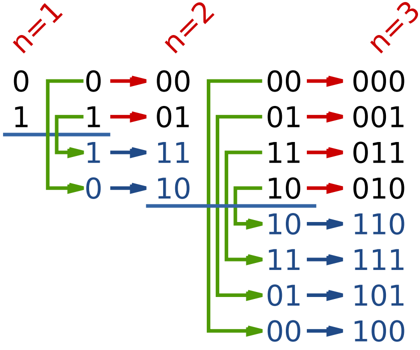
Binary to Gray:

```
(ctr >> 1) ^ ctr
```

ctr	bin	gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Technische Universiteit Eindhoven University of Technology

Gray Code



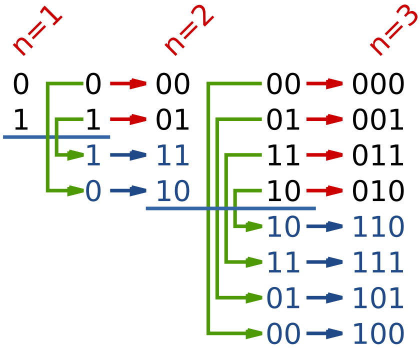
ctr	bin	gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Binary to Gray:

```
(ctr >> 1) ^ ctr
```


TU/e Technische Universiteit Eindhoven University of Technology

Gray Code

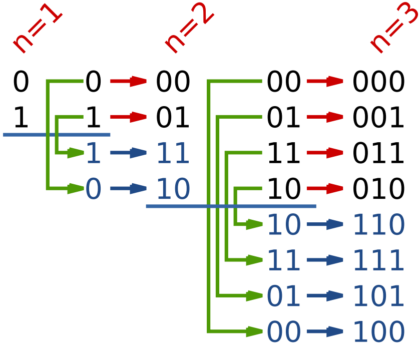


Binary to Gray:

```
(ctr >> 1) ^ ctr
```

ctr	bin	gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Gray Code



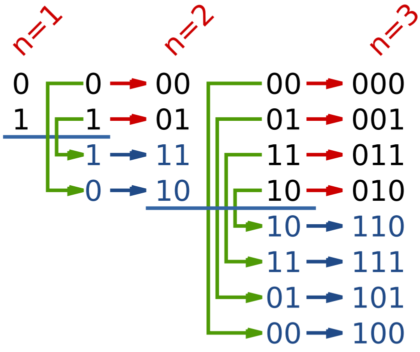
Binary to Gray:

```
(ctr >> 1) ^ ctr
```

ctr	bin	gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Technische Universiteit Eindhoven University of Technology

Gray Code



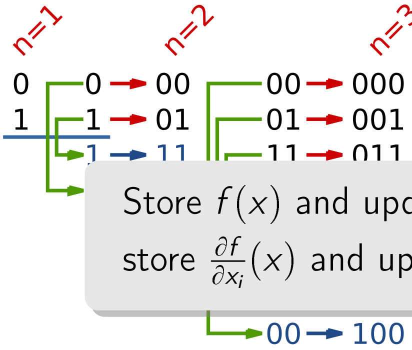
Binary to Gray:

```
(ctr >> 1) ^ ctr
```

ctr	bin	gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

TU/e Technische Universiteit Eindhoven University of Technology

Gray Code



Store $f(x)$ and update using $\frac{\partial f}{\partial x_i}(x)$;
store $\frac{\partial f}{\partial x_i}(x)$ and update using $\frac{\partial^2 f}{\partial x_i \partial x_j}(x)$.

Binary to Gray:

```
(ctr >> 1) ^ ctr
```

ctr	bin	gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

TU/e Technische Universiteit Eindhoven University of Technology

Gray-Code Algorithm

```
function EVAL(s)
    while s.i < 2^n do
        s.i ← s.i + 1;
        k_1 ← BIT_1(s.i);
        k_2 ← BIT_2(s.i);
        if k_2 valid then
            s.d'[k_1] ← s.d'[k_1] ⊕ s.d''[k_1, k_2];
        end if
        s.y ← s.y ⊕ s.d'[k_1];
        if s.y = 0 then
            return shr(s.i, 1) ⊕ s.i;
        end if
    end while
end function
```

Parallelization

Fix i Variables for 2^i Parallel Instances:

$$f = x_4 x_2 + x_3 x_0 + x_2 x_1 + x_3 + x_1 + x_0 + 1$$

e.g. $i = 2$:

$$f_{00_b} = 0 \cdot x_2 + 0 \cdot x_0 + x_2 x_1 + 0 + x_1 + x_0 + 1$$
$$f_{01_b} = 0 \cdot x_2 + 1 \cdot x_0 + x_2 x_1 + 1 + x_1 + x_0 + 1$$
$$f_{10_b} = 1 \cdot x_2 + 0 \cdot x_0 + x_2 x_1 + 0 + x_1 + x_0 + 1$$
$$f_{11_b} = 1 \cdot x_2 + 1 \cdot x_0 + x_2 x_1 + 1 + x_1 + x_0 + 1$$

2^i independent equations (systems)

Technische Universiteit Eindhoven University of Technology

Parallelization

Fix i Variables for 2^i Parallel Instances:

$$f = x_4x_2 + x_3x_0 + x_2x_1 + x_3 + x_1 + x_0 + 1$$

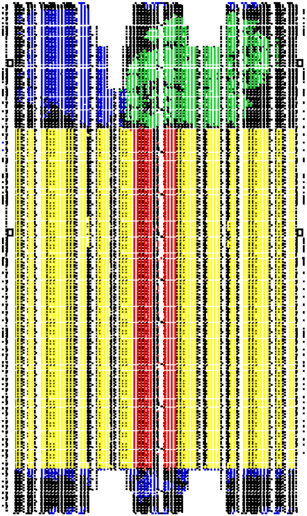
e.g. $i = 2$:

$$\begin{aligned}
f_{00_b} &= 0 \cdot x_2 + 0 \cdot x_0 + \boxed{x_2x_1} + 0 + x_1 + x_0 + 1 \\
f_{01_b} &= 0 \cdot x_2 + 1 \cdot x_0 + \boxed{x_2x_1} + 1 + x_1 + x_0 + 1 \\
f_{10_b} &= 1 \cdot x_2 + 0 \cdot x_0 + \boxed{x_2x_1} + 0 + x_1 + x_0 + 1 \\
f_{11_b} &= 1 \cdot x_2 + 1 \cdot x_0 + \boxed{x_2x_1} + 1 + x_1 + x_0 + 1
\end{aligned}$$

2^i independent equations (systems)
sharing the *same* quadratic terms!

Technische Universiteit Eindhoven University of Technology

System Solving — Brute Force



80-bit Security:

Solving a system of 80 variables requires 1042 days on 65,536 Spartan-6 FPGAs at a total cost of about US$40 million.