New results on symmetric quantum cryptanalysis

María Naya-Plasencia

New Results on Symmetric Quantum Cryptanalysis

María Naya-Plasencia

Inria, France

ERC project QUASYModo

Workshop on Flexible Symmetric Cryptography

Lorentz Center- 21 Mars 2018

Outline

- Introduction
  On Quantum-Safe Symmetric Cryptography

- Efficient Quantum Collision Search
  joint work with A. Chailloux and A. Schrottenloher
  [Asiacrypt17]

- On Modular Additions
  joint work with X. Bonnetain

Symmetric Cryptography

Classical Cryptography

Enable secure communications even in the presence of malicious adversaries.

Asymmetric (e.g. RSA) *(no key exchange/computationally costly)*

Security based on well-known hard mathematical problems (e.g. factorization).

Symmetric (e.g. AES) *(key exchange needed/efficient)*

Ideal security defined by generic attacks ($2^{|K|}$).

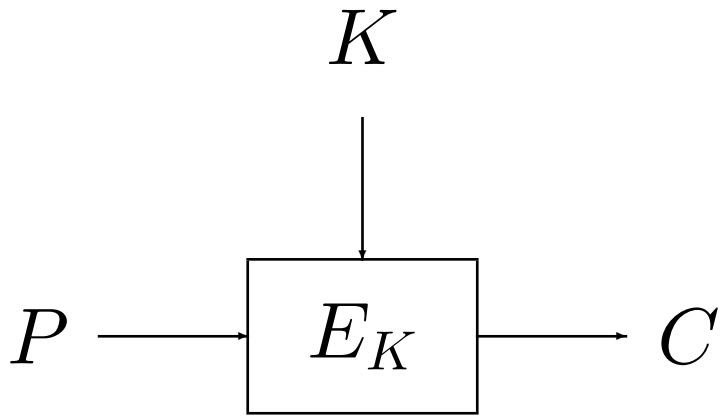
Need of continuous security evaluation (cryptanalysis).

$\Rightarrow$ Hybrid systems! (e.g. in SSH)

Symmetric primitives

- Block ciphers, (stream ciphers, hash functions..)

Message decomposed into blocks, each transformed by the same function E_K.



E_K is composed of a round transform repeated through several similar rounds.

Generic Attacks on Ciphers

- Security provided by an ideal block cipher defined by the best generic attack:
  exhaustive search for the key in $2^{|K|}$.

- Recovering the key from a secure cipher must be infeasible.

  $\Rightarrow$ typical key sizes $|K| = 128$ to 256 bits.

Cryptanalysis: Foundation of Confidence

> Any attack better than the generic one
> is considered a "break".

- Proofs on symmetric primitives need to make unrealistic assumptions.
- We are often left with an empirical measure of the security: cryptanalysis.
- Security redefinition when a new generic attack is found (e.g. accelerated key search with bicliques [BKR 12])

Current scenario

- Competitions (AES, SHA-3, eSTREAM, CAESAR).
- New needs: lightweight, FHE-friendly, easy-masking.
  $\Rightarrow$ Many good proposals/candidates.

- How to choose?

- How to be ahead of possible weaknesses?

- How to keep on trusting the chosen ones?

Cryptanalysis: Foundation of Confidence

When can we consider a primitive as secure?

- A primitive is secure as far as no attack on it is known.
- The more we analyze a primitive without finding any weaknesses, the more reliable it is.

Design new attacks + improvement of existing ones:

- essential to keep on trusting the primitives,
- or to stop using the insecure ones!

On weakened versions

If no attack is found on a given cipher, what can we say about its robustness, security margin?

The security of a cipher is not a 1-bit information:

- Round-reduced attacks.
- Analysis of components.

$\Rightarrow$ determine and adapt the security margin.

On high complexities

When considering large keys, sometimes attacks breaking the ciphers might have a very high complexity far from practical *e.g.*. 2^{120} for a key of 128 bits.

Still dangerous because:

- Weak properties not expected by the designers.
- Experience shows us that attacks only get better.
- Other existing ciphers without the "ugly" properties.

- When determining the security margin: find the highest number of rounds reached.

Post-Quantum Symmetric Cryptography

Post-Quantum Cryptography

Adversaries have access to **quantum computers.**

Asymmetric (e.g. RSA):

Shor's algorithm: Factorization in polynomial time

$\Rightarrow$ current systems not secure!

Solutions: lattice-based, code-based cryptography...

Symmetric (e.g. AES):

Grover's algorithm: Exhaustive search from $2^{|K|}$ to $2^{|K|/2}$.

Double the key length for equivalent ideal security.

We don't know much about cryptanalysis of current ciphers when having quantum computing available.

Post-Quantum Cryptography

Problem for present existing long-term secrets.
$\Rightarrow$ start using quantum-safe primitives NOW.

Important tasks:

- Conceive the cryptanalysis algorithms for evaluating the security of symmetric primitives in the P-Q world.

- Use them to evaluate and design symmetric primitives for the P-Q world.

Quantum Symmetric Cryptanalysis

Some recent results on Q-symmetric crytanalysis:

3-R Feistel [Kuwakado-Morii10], Even-Mansour [Kuwakado-Morii12], Mitm [Kaplan14], Related-Key [Roetteler-Steinwandt15], Diff-lin [Kaplan-Leurent-Leverrier-NP16], Simon's[Kaplan-Leurent-Leverrier-NP16], FX [Leander-May17], parallel multi-preim. [Banegas-Bernstein17], Multicollision [Hosoyamada-Sasaki-Xagawa17],
AEZ [Bonnetain17]...

Quantum Symmetric Cryptanalysis

Two main models used:

- Q1:
  classical queries and access to a quantum computer.

- Q2:
  +superposition queries to a quantum cryptog. oracle.

Very powerful, BUT...

Q2: Superposition Model

Many good reasons to study security in this scenario:

- Simple
- Non-trivial: Many constructions still seem resistant: AES, SALSA20, NMAC, HMAC...
- Inclusive of all intermediate scenarios

Defined and used in: [Zhandry12], [Boneh-Zhandry13], [Damgård-Funder-Nielsen-Salvail13], [Mossayebi-Schack16], [Song-Yun17], Simon's attacks, FX, AEZ...

An attack in this model $\Rightarrow$ not safe to implement the primitive in a quantum computer.

On Quantum attacks

- Compare to best generic attack,
- generic attack is accelerated, so
- broken classical primitive might be unbroken in a quantum setting.

Collision Search

w. A. Chailloux & A. Schrottenloher

Collision Search Problem

Given a random function $H : \{0,1\}^n \to \{0,1\}^n$, find $x, y \in \{0,1\}^n$ with $x \neq y$ such that $H(x) = H(y)$.

Many applications: *i.e.* generic attacks on hash functions.

(Multi-preimage search can be seen as a particular case).

Best known algorithms

	Time	Queries	Memory
Pollard's rho	$2^{n/2}$	$2^{n/2}$	$poly(n)$
Parallelization (2^s)	$2^{n/2-s}$	$2^{n/2}$	2^s

	Time	Queries	Qubits
Grover	$2^{n/2}$	$2^{n/2}$	$poly(n)$
BHT	$2^{2n/3}*$	$2^{n/3}$	$poly(n)*$
Ambainis	$2^{n/3}$	$2^{n/3}$	$2^{n/3}$

Open Questions

Challenge 1: Find an algorithm for collision and/or element distinctness which gives a searching speedup greater than merely a square-root factor over the number of available processing qubits[a]

[a] Grover and Rudolph, *How significant are the known collision and element distinctness quantum algorithms?* 2004.

Considered Model

- The same one as in all the previous quantum algorithms BUT we limit the amout of quantum memory available to a small amount $poly(n)$.

- Available small quantum computers seems like the most plausible scenario.

- We are interested in the theoretical algorithm and we did not take into account implementation aspects.

Starting Point: BHT Algorithm

- Optimal number of queries,
- $poly(n)$ qbits,
- But time?

BHT: Summarized procedure

- Build a list L of size $2^{n/3}$ elements (classic memory),

- Exhaustive search for finding one element that collides: With AA, the number of iterations is $(\frac{2^n}{2^{n/3}})^{1/2} = 2^{n/3}$.

Testing the membership with L for the superposition of states costs $2^{n/3}$ with n qbits:

$$\text{Time: } 2^{n/3} + 2^{n/3}(1 + 2^{n/3}) \approx 2^{2n/3}$$

Can we improve this?

Lets build the list L with distinguished points
e.g. $H(x_i) = 0^u || z$, for $z \in \{0,1\}^{n-u}$.

The cost of building the list is bigger: $2^{n/3+u/2}$.
The setup of AA is bigger: $2^{u/2}$
The membership test stays the same: $|L| = 2^{n/3}$
BUT The number of iterations is smaller: $2^{n/3-u/2}$

Time: $2^{n/3+u/2} + 2^{n/3-u/2}(2^{u/2} + 2^{n/3}) \approx 2^{2n/3-u/2} + 2^{n/3+u/2}$

With optimal parameters

The cost will be optimized for a certain size of L: $2^v \neq 2^{n/3}$.

Time: $2^{v+u/2} + 2^{\frac{n-v-u}{2}}(2^{u/2} + 2^v)$

For $v = n/5$, $u = 2n/5$: Time: $\tilde{\mathrm{O}}(2^{2n/5})$

For multiple preimage search, the algorithm is similar, but we only keep in L the distinguished points amongst the already given ones.

Comparison

	Time	Queries	Qubits	Classic Memory
Pollard	$2^{n/2}$	$2^{n/2}$	0	$poly(n)$
Grover	$2^{n/2}$	$2^{n/2}$	$poly(n)$	0
BHT	$2^{2n/3}$	$2^{n/3}$	$poly(n)$	$2^{n/3}$
Ambainis	$2^{n/3}$	$2^{n/3}$	$2^{n/3}$	0
New algorithm	$2^{2n/5}$	$2^{2n/5}$	$poly(n)$	$2^{n/5}$

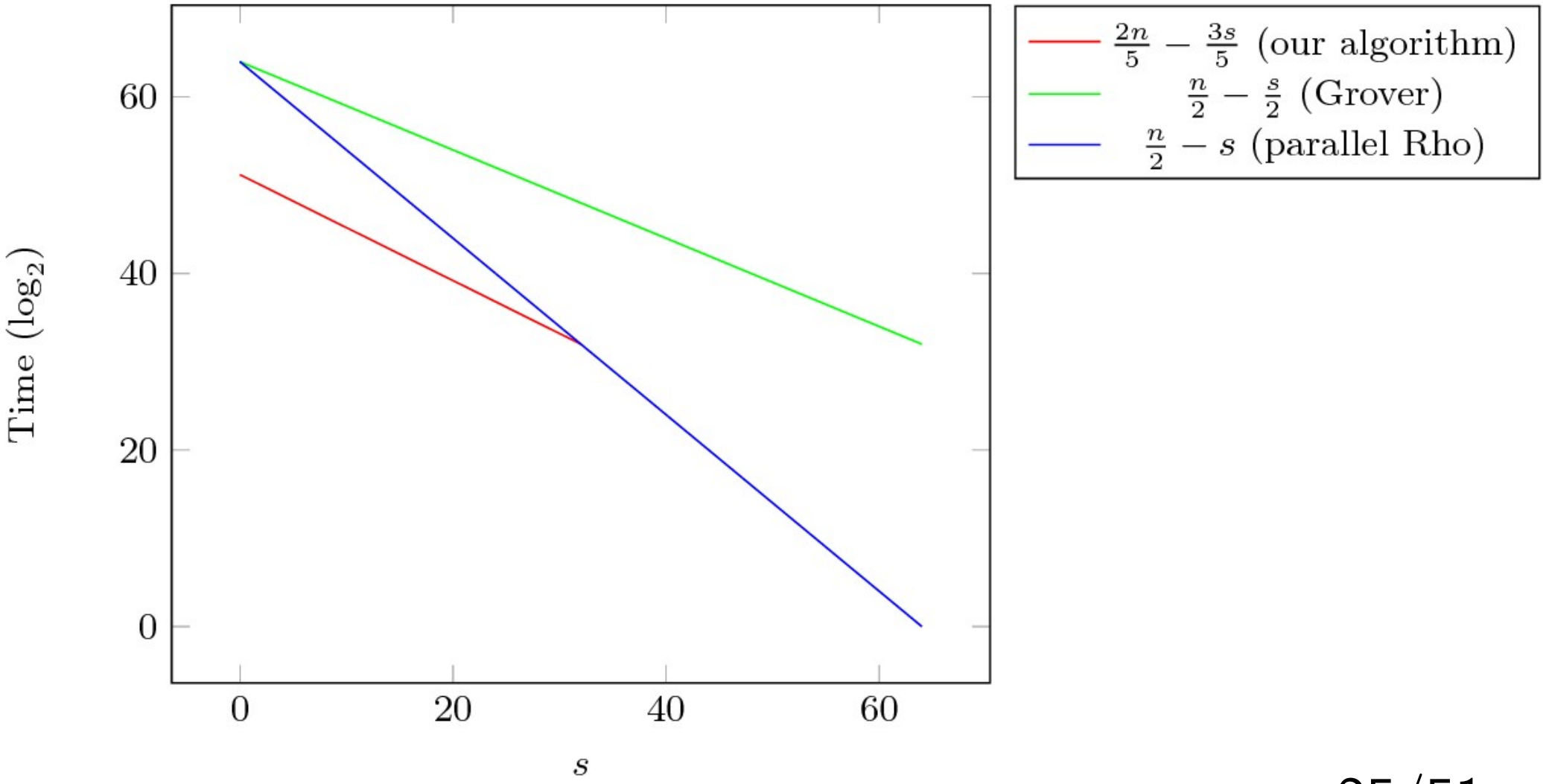
Parallelization

With 2^s n-qbit registers and "external" parallelization we can achieve:

Time: $2^{v+u/2-s} + 2^{\frac{n-v-u}{2}-s/2}(2^{u/2} + 2^v)$

Our theoretical algorithm seems more efficient than classical parallelization/Beal up to $s = n/4$

Comparison example: n=128



Example of Applications (1)

- 1. Hash functions: Collision and Multi-preimages time from $2^{n/2}$ to $2^{2n/5}$ and $2^{3n/7}$ (Q1).
  Ex.- time and queries for $n = 128$:
  rho$= 2^{64}$, ours$= 2^{51.2}$ (with less than 1GB classical)

- 2. Multi-user setting: Recover Ctxt, from same Ptxt, 2^t different keys: apply multi-preimage algorithm (Q1).
  Depending on the value of t different gain.

Example of Applications (2)

- 3. Operation modes: Collision attacks on CBC: 2^t Ctxt, find one preimage $\Rightarrow$ Ptxt. (Q2). If frequent rekeying (Q1).
- 4. Bricks for Cryptanalysis: Collision, multi-preimage search: often bricks of more technical cryptanalysis: improve the steps.

Conclusion 1

We solved challenge 1 for Grover and Rudolph 2004: new efficient collision search algorithm with small quantum memory.

Many applications in symmetric cryptograhy.

Open question: is it possible to meet the optimal $2^{n/3}$ in time with small quantum memory? (Quantum random walks, quantum learning graphs...?)

On Modular Additions

with X. Bonnetain

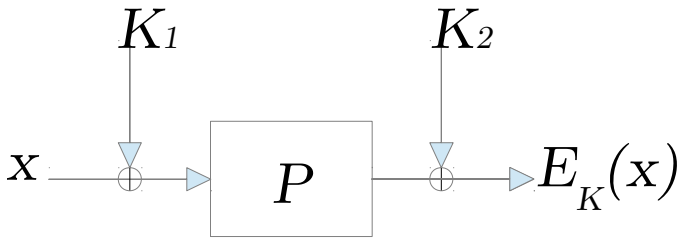
Quantum cryptanalysis: Simon's algorithm

Simon's problem: Given $f : \{0,1\}^n \to \{0,1\}^n$ such that $\exists s \mid f(x) = f(y) \iff [x = y \text{ or } x \oplus y = s]$, find s.

- Classical complexity: $\Omega(2^{n/2})$.
- Quantum complexity [Simon 94]: $O(n)$.

Simon's algorithm in Symmetric Cryptography

- Even-Mansour cipher [Even Mansour 97]: $DT > 2^n$

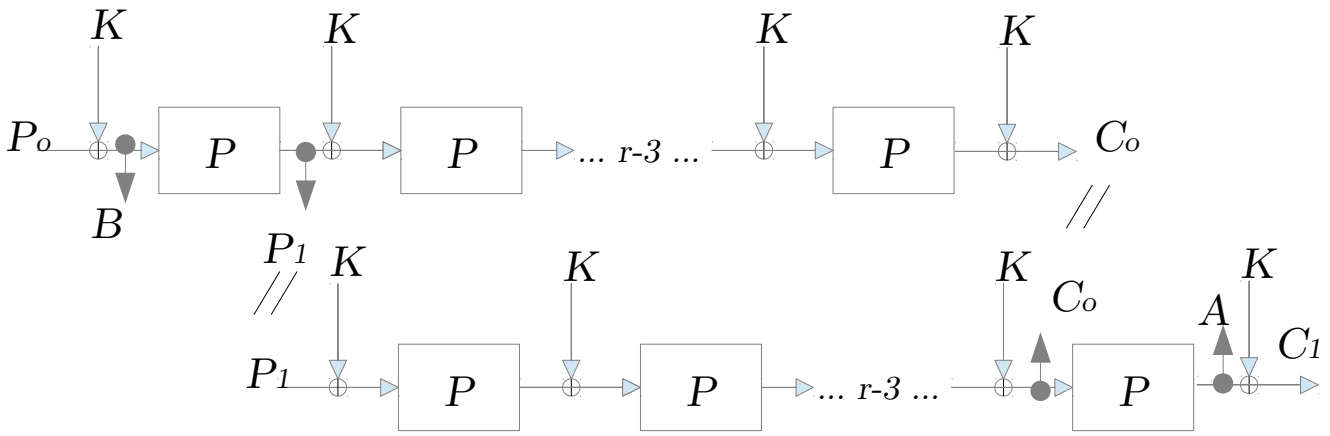


$f(x) = E_K(x) \oplus P(x) \rightarrow f(x) = f(x \oplus K_1)$
Simon's algo. on $f \Rightarrow K_1$ in $\mathcal{O}(n)$ [Kuwakado Morii 12] (Q2)

- Related-key attacks [Roetteler Steinwandt 15]
- 3-round Feistel [Kuwakado Morii 10]
- LWR, modes of operation for authentication (CBC-MAC, PMAC, OCB..), some CAESAR candidates [KLLN-P 16b]

Simon's algorithm and Slide attacks

- Classical: $\mathcal{O}(2^{n/2})$ [Biryukov Wagner 99]



- Quantum: Simon $\mathcal{O}(n)$ [KLLN-P 16b]

$$f : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$$

$$b, x \quad \mapsto \begin{cases} P(E_K(x)) \oplus x & \text{if } b = 0, \\ E_K(P(x)) \oplus x & \text{if } b = 1. \end{cases}$$

$f(x) = f(x \oplus (1||K))$

Simon's algorithm in Symmetric Cryptography

Some (NOT ALL) primitives secure in the classical world become completely broken in the Q2 model.

This implies that it is not safe to implement such devices on quantum computers.

This does not seems to imply that these primitives are unsafe in any other setting.

Tweaking to resist Simon's algo. in Q2?

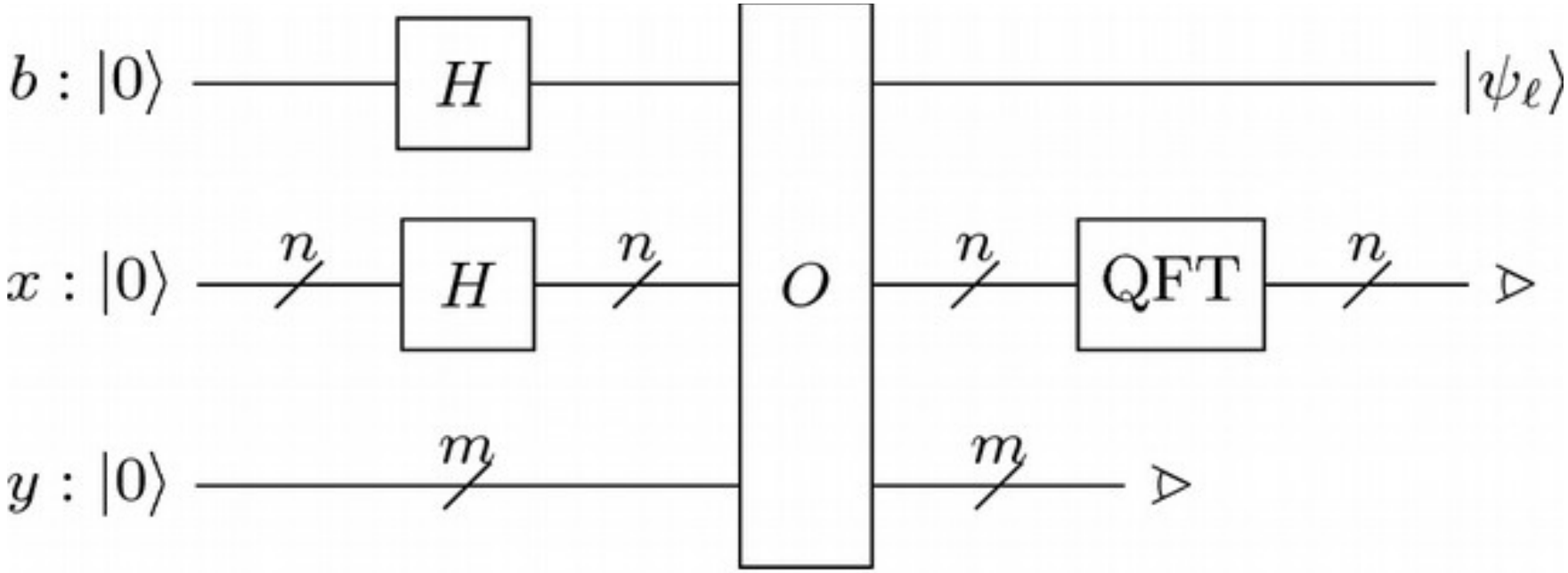
- In [Alagic Russell 17] several proposals. Most efficient: replace xor by modular additions.
- Hidden shift problem in $\mathbb{Z}/(N)$.
- No algorithm in polynomial time: Kuperberg in $2^{O(\sqrt{n})}$
- Up to what point do primitives resist?

Motivation and results

- 5. Dimensionate symmetric primitives
- 1. More precise evaluation of Kuperberg's algorithm complexity+improvement
- 2. Example of application with Poly1305
- 3. What about parallel modular additions?
- 4. New Quantum attacks (Feistel's slide, FX)

Kuperberg's Algorithm[2003]

$f_0(x) = f_1(x + s)$



$|\psi_\ell\rangle = |0\rangle + \exp\left(2\iota\pi\frac{s\ell}{2^n}\right)|1\rangle$

Kuperberg's Algorithm[2003]

When $\ell = 2^{n-1}$, we can recover the parity of s:
$|\psi_{2^{n-1}}\rangle = |0\rangle + \exp(\iota\pi s)\,|1\rangle$

1) Recover many $|\psi_{\ell_i}\rangle$.
2) Combine pairs $|\psi_{\ell_1}\rangle$ and $|\psi_{\ell_2}\rangle$ with a CNOT:
   $\text{CNOT}\,|\psi_{\ell_1}\rangle\,|\psi_{\ell_2}\rangle = |\psi_{\ell_1+\ell_2}\rangle\,|0\rangle + \chi(\frac{s\ell_2}{2^n})\,|\psi_{\ell_1-\ell_2}\rangle\,|1\rangle$
3) Measure 2nd qbit: if 0 we have $|\psi_{\ell_1+\ell_2}\rangle$, if 1 $|\psi_{\ell_1-\ell_2}\rangle$.

Combining elements with the same divisibility by two we get closer to $\ell = 2^{n-1}$.

Kuperberg's Simulation

Generate N random numbers in $\mathbb{Z}/(2^n)$
Separate them in pools P_i of elements divisible by 2^i and not 2^{i+1}
for $i := 0$ to $n-2$ **do**
    while $|P_i| \geq 2$ **do**
        Pop two elements (a, b) of P_i where $a+b$ or $a-b$ has the highest possible divisibility by 2 (and is not 0)
        c is chosen randomly in $\{a+b, a-b\}$
        Insert c in the corresponding P_j
        if $P_{n-1} \neq \emptyset$ **then return** Found
        end if
    end while
end for
return Failure

Improvement and Simulation

- Our improvement: all the bits with one iteration. $O(n^2 2^{\sqrt{2\log_2(3)n}}) \Rightarrow O(n 2^{\sqrt{2\log_2(3)n}})$

- Our simulations give: $0.7 \times 2^{1.8\sqrt{n}}$ for recovering full s. Code available: ask Xavier Bonnetain if interested.
  `xavier.bonnetain@inria.fr`

Application example with Poly1305

Poly1305 in the Q2 model: *superposition*-Poly1305.

Two 128-bit keys (r, k), 128-bit nonce n, message m array of 128-bit blocks, output 128-bit tag.
Poly1305-AES$_{(r,k,n)}(m_1, \ldots, m_q) =$
$\left(\sum_{i=1}^{q}(m_{q-i+1} + 2^{128})r^i \mod (2^{130} - 5)\right) + \mathsf{AES}_k(n)$

Access to:
$Poly_n^2 : |m_1\rangle\,|m_2\rangle\,|0\rangle \mapsto |m_1\rangle\,|m_2\rangle\,\left|\text{Poly1305-AES}_{(r,k,n)}(m_1, m_2)\right\rangle,$

Superposition-Poly1305

We denote

$$F(x) = \text{Poly1305-AES}_{(r,k,n)}(1, x)$$
$$= \left(f(x) \mod (2^{130} - 5)\right) + \text{AES}_k(n) \text{ and}$$
$$G(x) = \text{Poly1305-AES}_{(r,k,n)}(0, x)$$
$$= \left(g(x) \mod (2^{130} - 5)\right) + \text{AES}_k(n),$$

which satisfy, for the same nonce, $F(x) = G(x + r)$.

As $f(x) = xr + r^2 + 2^{128}(r + r^2))$, $g(x) = xr + 2^{128}(r + r^2)$ and $f(x) = g(x + r)$.

Apply Kuperberg to find the hidden shift r.

Superposition-Poly1305

Two issues:

- One nonce, one query to both $F(x)$ and $G(x)$:
  we can compute $(1, x)$ and $(0, x)$ in superposition in one register and call the oracle $Poly_n^2$ on it.
- We cannot sample all group elements: consider 2^{18} possible intervals for r of size 2^{106}:
  $r \in [2^{106}c, 2^{106}(c+1))$ for $c \in [0, 2^{18})$ and the functions $f(x)$ and $g(x + 2^{106}c)$. Bad element with pb 2^{-21}.
  Apply Kuperberg to each interval: 2^{20}.
  Complexity: 2^{39} for r (thanks to our improvement!).

Algorithm for Parallel Modular Additions?

- HSP problem for groups product of cyclic groups
- Recurrent problem in symmetric cryptography
- Kuperberg not optimal

Simon meets Kuperberg

Algorithm for solving the case of p modular additions of words of w, matching Simon's $(w = 1)$ and Kuperberg's $(p = 1)$

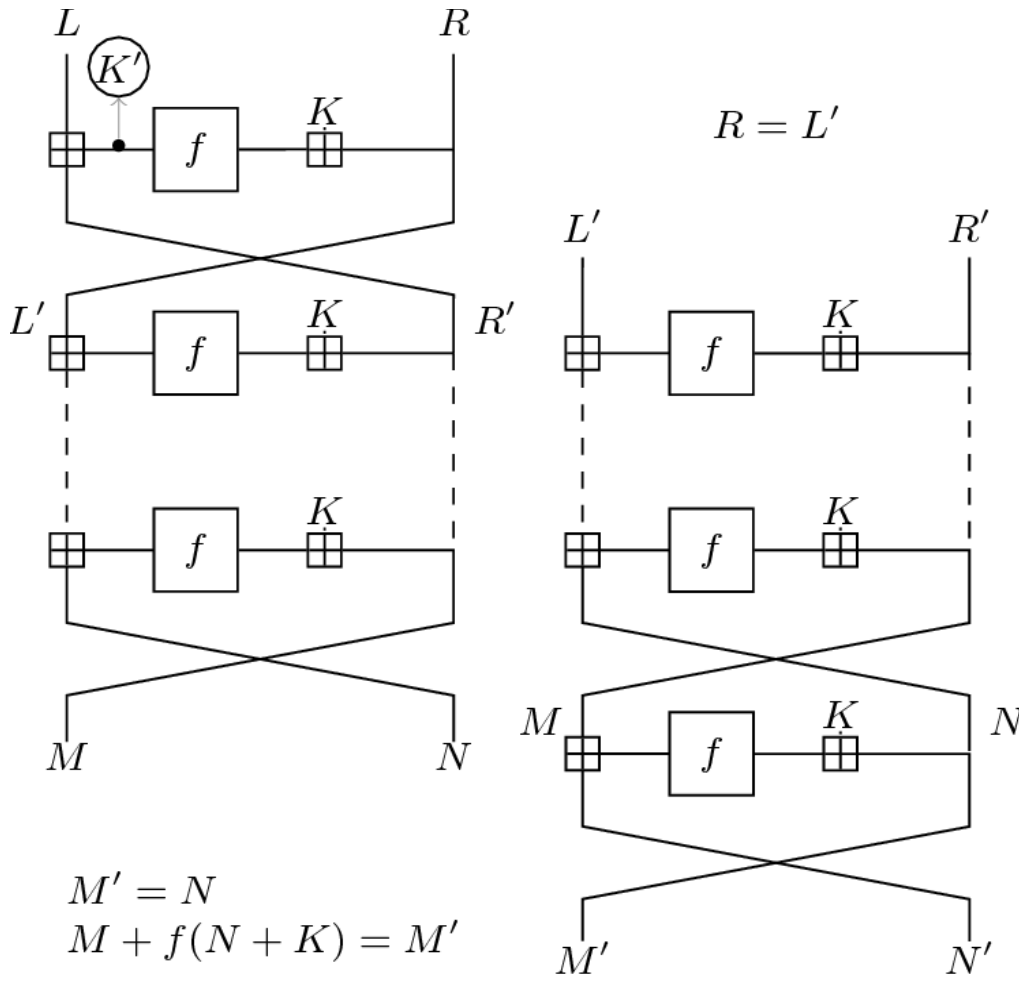
- First Idea: Kuperberg's variant- better worst-case gain
- Second Idea: $p + 1$ equations always gain p zeros
- Combining both: best method depends on parameters and thresholds.

New Quantum Attacks

- Advance slide attacks on Feistel ciphers
- Attacks on Feistel ciphers with non-invertible functions
- FX construction (quantum [Leander-May17]) with modular additions

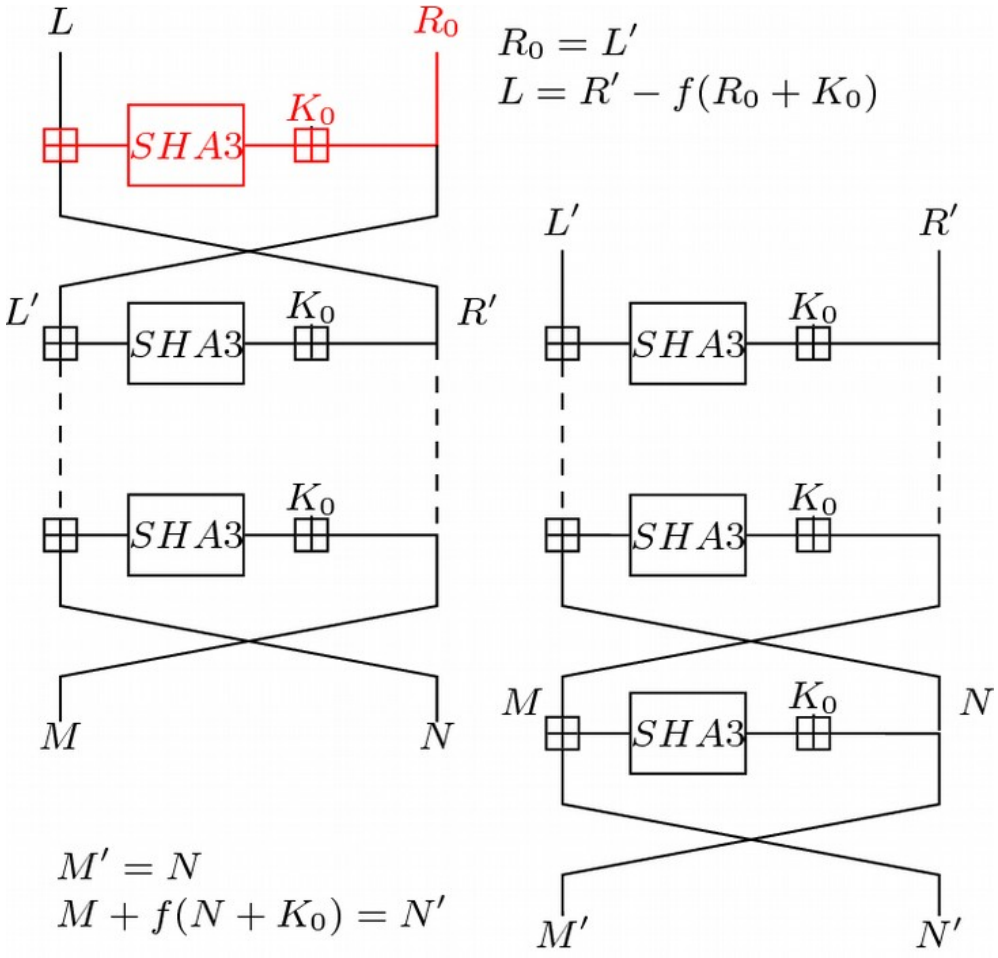
Feistel with 1 key

R_0 fixed: $E(x, R_0)_R = E(R_0, x + f(R_0 + K))_L$



Recover $K' \to K$ in $2^{1.2\sqrt{n}}$.

Feistel with non-invertible f



F takes R_i as an input and outputs $K' = \mathsf{SHA3}(R_i + K)$,

$F(x) = \mathsf{SHA3}(x + K)$. All key and branch combinations.

Needed Sizes?

	Even-Mansour	LRW	Op. Modes
State (bits)	5200	5200	5200
Key (bits)	5200	$k \geq 256$	$k \geq 256$

Table 1: Summary of constructions and parameters in order to provide 128-bit security when using modular additions instead of $\oplus$.

Needed Sizes?

	Key-Alt.	2k-DES$\oplus$	2k-DES+	FX$\oplus$	FX+
State	5200	2^{127}	5200	240	205
Key	5200	2^{127}	5200	480	410

Table 2: Summary of constructions and sizes for 128-bit security in order to resist to the best known quantum attacks.

Conclusion 2

- Improved Kuperberg's algorithm and new algorithm for parallel modular additions.
- State size needed for a 128-bit security.
  at least 5200 bits (but for FX) $\Rightarrow$ not very realistic.
- Might be better to just avoid vulnerable constructions, or try different patches (if we are concerned by superposition attacks).
- *Superposition*-Poly1305 broken implies that Poly1305 should not be implemented in a Quantum computer.

Final Conclusion

Open problems

- Optimal collision time $2^{n/3}$?.
- $\alpha-$XOR problem $\Rightarrow$ started, improvement this week?.
- Algebraic attacks $\Rightarrow$ this week?
- Boomerang attacks $\Rightarrow$ this week?
- FSE Stevens: Quantum cryptanalysis of SHA-2?
- AES quantum evaluation- on going work.
- Generic key-length extensions?
- What about state size? ...

Symmetric Quantum Cryptanalysis[1]

Lots of things to do !

[1]Thanks to X. Bonnetain, A. Chailloux and A. Schrottenloher for their help with the slides