How to Improve Rebound Attacks

María Naya-Plasencia
FHNW - Switzerland

Outline

1. Hash Functions and the SHA-3 Competition

2. The Rebound Attack and Motivation

3. Merging Lists with Respect to t
   - Problem 1
   - Problem 2

4. Results and Conclusion

Hash Functions and the SHA-3 Competition

Cryptographic Hash Functions

$$\mathcal{H} : \{0,1\}^* \longrightarrow \{0,1\}^{\ell_h}$$

- Given a message of arbitrary length returns a short 'random-looking' value of fixed length.
- Many applications: MAC's (authentication), digital signatures, integrity check of executables, pseudo - random generation...

Hash Function Security Requirements

- Classical and main security requirements: collision resistance and (second) preimage resistance.
- Other types of attacks: near-collisions, multicollisions, length extension attacks, distinguishers...
- Security proofs rely on assumptions on the building blocks: *i.e.*, ideal permutation, collision-resistant compression function... $\Rightarrow$ "attack the assumptions".

NIST [1] SHA-3 Competition

- Attacks known for current standards MD5 and SHA-1 [Wang-Yu 05, Wang et al. 05].
- Confidence in SHA-2 (standard) undermined.
- NIST has launched the SHA-3 public competition for finding a new hash standard.

[1] U.S. Institute of Standards and Technology

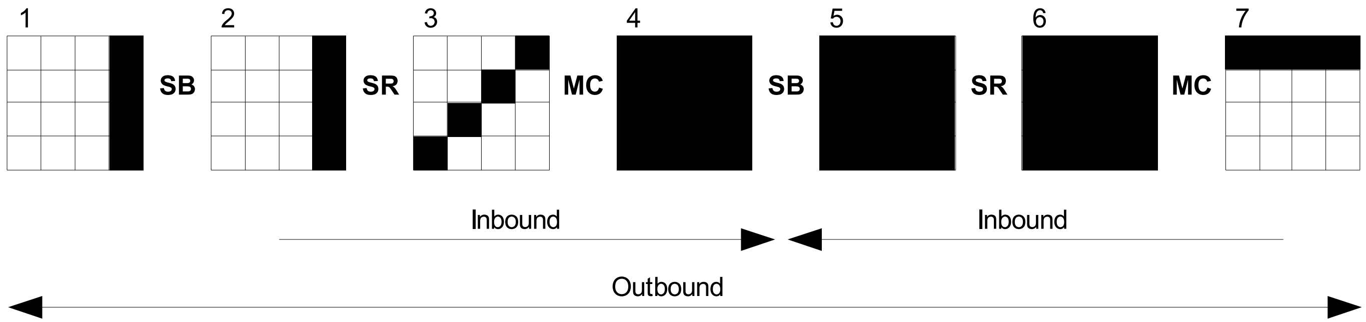
NIST SHA-3 Competition

- 64 submissions (October 2008).
- 51 first round candidates (October 2008).
- 14 second round candidates (July 2009).
- 5 finalists (December 2010).

- NIST will choose the new hash function standard in 2Q 2012.

The Rebound Attack and Motivation

Rebound Attack [Mendel et al.09]



Inbound phase:

1. We choose the differential path,
2. we find differences for the black bytes that verify the path with a meet in the middle (probability=2^{-16}),
3. then, for each difference match, 2^{16} values make the path possible.

Rebound Attack

- Low cost solutions for a low probability part of the path.

- At first introduced for analysing AES-based functions.

- Improvements: multi-inbounds [Matusiewicz et al.09], super-sboxes [Gilbert-Peyrin10, Lamberger et al.09]...
  $\Rightarrow$ Quite technical.

- Applied to several SHA-3 candidates to build: collisions, semi-free-start collisions, distinguishers...

The Rebound Attack Applied to SHA-3:

1. ECHO
2. Grøstl
3. JH
4. Luffa
5. LANE
6. Shavite
7. Cheetah (simple and low complexity)
8. Twister (simple and low complexity)
9. Skein (high level)

We Have Noticed that...

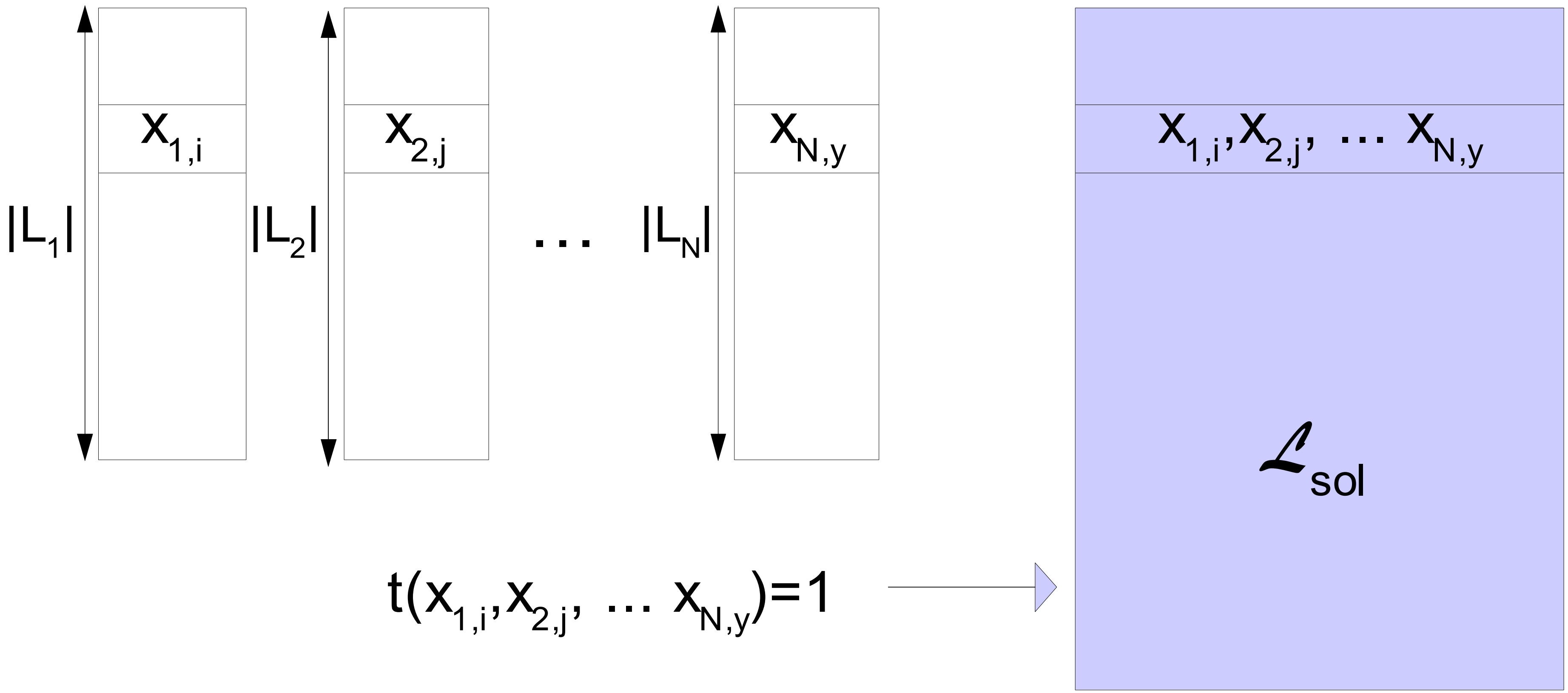
- In nearly all the cases, a *merge* of big lists is needed,
- and that is very often not done in an optimal way.

We Propose

- Some problem definitions that will help improving the complexities.
- Some algorithms for solving these problems.
- The main aim is to help future rebound attacks to be as efficient as possible.

Merging N Lists with Respect to t

General Problem

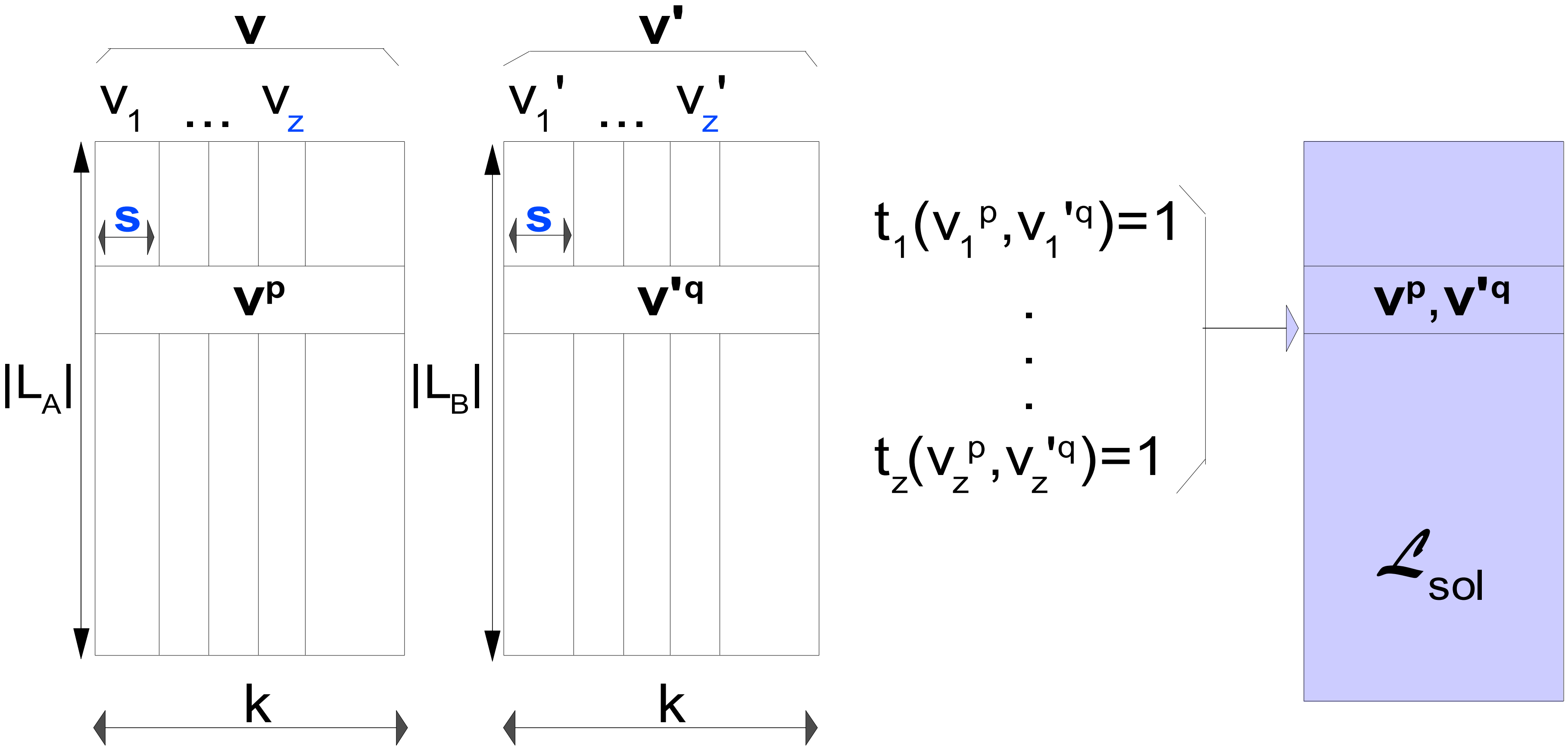


A priori, complexity of merging the N lists with respect to t :

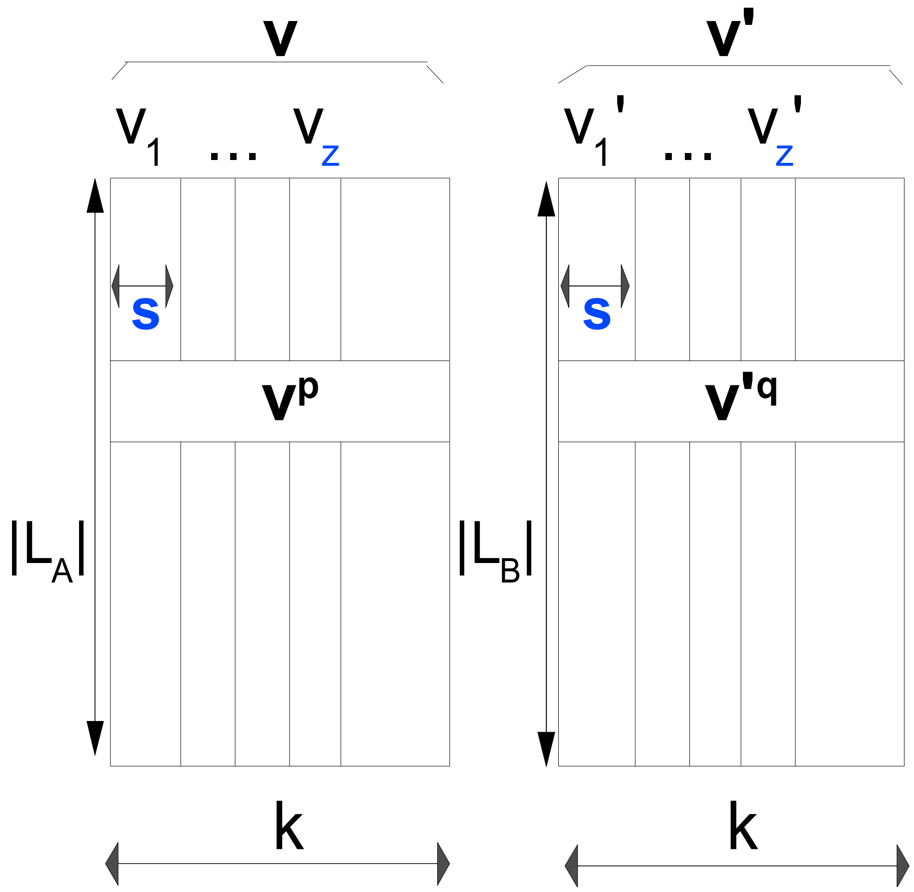
$|L_1| x |L_2| x \ldots x |L_N|$

Problem 1: Group-Wise t

It can be reduced to a $N = 2$ situation with L_A and L_B.



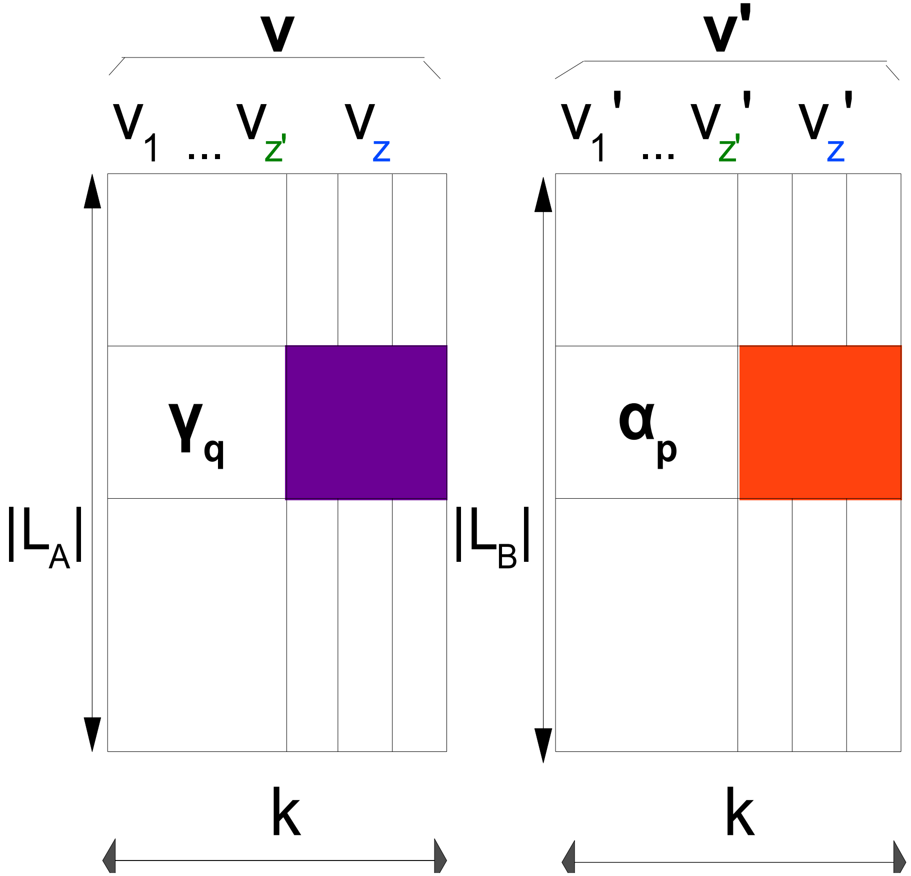
Solving Problem 1: Instant Matching



How many elements can be associated to $\mathbf{v'}^q$ by t? $P_t 2^{zs}$.

If $P_t 2^{zs} < |L_A|$, instant matching provides better complexity than exhaustive search.

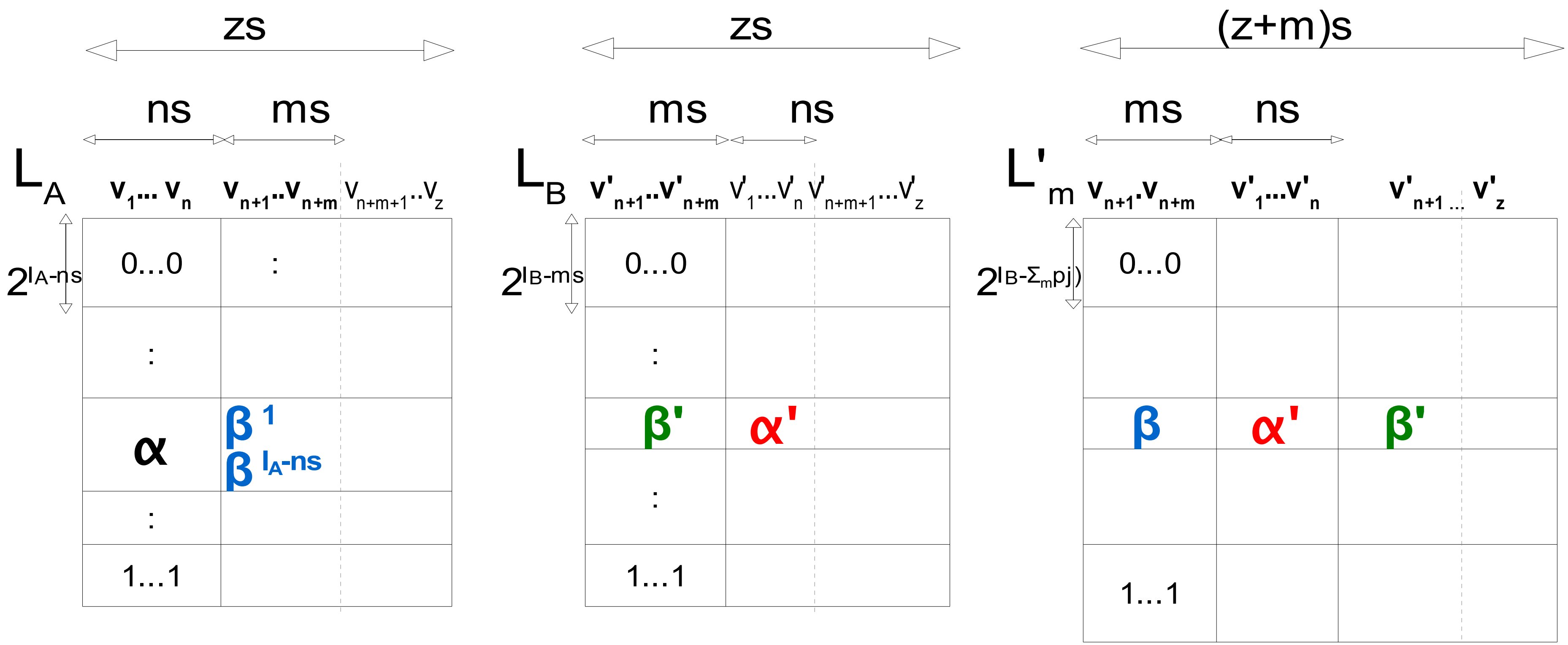
Solving Problem 1: Gradual Matching



If $P_t 2^{zs} > |L_A|$, choose z'<z and match the z' first groups.

Then, merge $L_B(\alpha_p)$ with $L_A(\gamma_q)$ for each match with respect to the (z-z') remaining groups.

Solving Problem 1: Parallel Matching

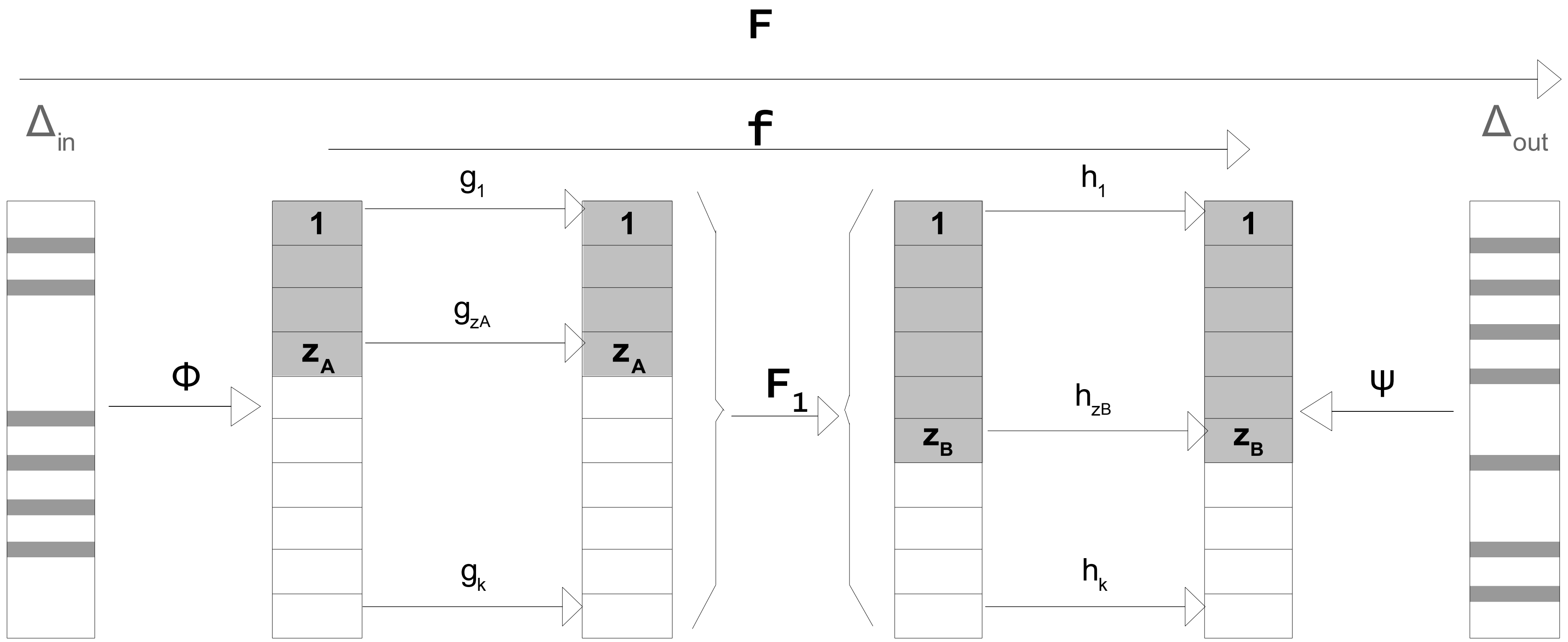


Building the auxiliary list L'_m allows us to match in parallel the first n groups (**α** and **α** ') and the m following ones (**β** and **β** ').

Problem 1: 3 Algorithms

Type of Matching	Time	Memory
Instant	$\mathcal{O}(z2^s + zP_t 2^{l_B + zs})$	$\mathcal{O}(z2^s + 2^{l_A} + 2^{l_B} + P_t 2^{l_A + l_B})$
Gradual (z' first groups)	$\mathcal{O}(z2^s + 2^{z's}(z' + \mathcal{S}2^{\text{merge}}))$	$\mathcal{O}(z2^s + 2^{l_A} + 2^{l_B} + \mathcal{S} + P_t 2^{l_A + l_B})$
Parallel (m and n groups in parallel)	$\mathcal{O}(2^{l_n} + 2^{l_m} + 2^{l_A + l_B - \sum_{j=1}^{n+m} p_j} + 2^{l_A + ns - \sum_{j=1}^{n} p_j} + 2^{l_B + ms - \sum_{j=n+1}^{m} p_j})$	$\mathcal{O}(2^{l_n} + 2^{l_m} + 2^{l_B} + 2^{l_B + ms - \sum_{j=n+1}^{m} p_j} + P_t 2^{l_A + l_B})$

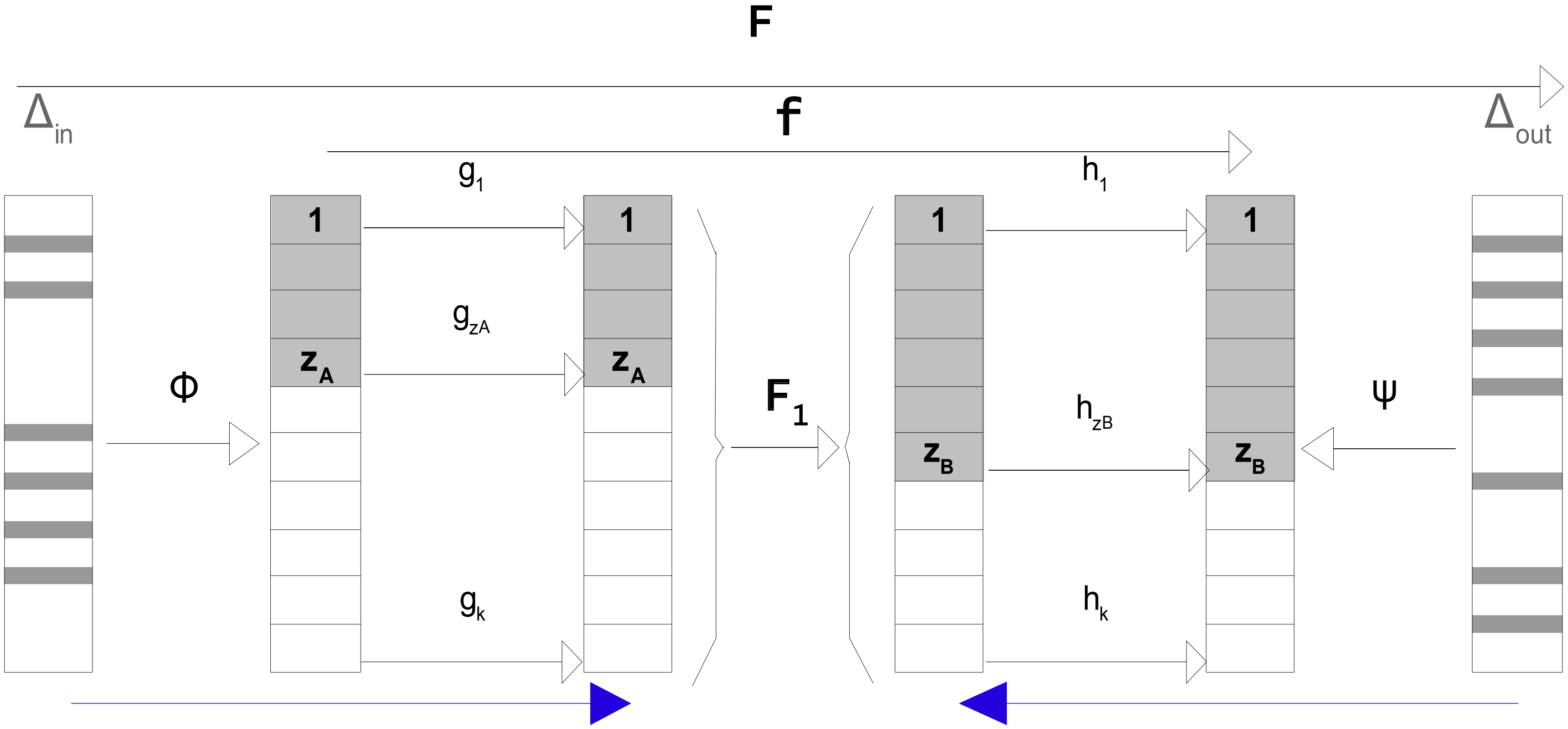
Problem 2: Parallel AES States



For all possibles Δ_{in} and Δ_{out}, find all x such that

$$F(x) \oplus F(x \oplus \Delta_{in}) = \Delta_{out}.$$

Problem 2: Stop-in-the-Middle



2) For a Δ_{in} fixed, compute the 2^s possible Z_A groups (L_i).

1) For all $\Delta_{out,}$ compute the 2^s possible Z_B groups ($L_{j,b}$).

3) Match the lists with F_1.

4) Check if all ($L_{j,b}$) belong to the same Δ_{out}.

s: size of the input/output of g_i and h_i.

The Rebound Attack Applied to SHA-3:

Out of the studied analysis, we have been able to improve the rebound attacks on:

1. ECHO
2. Grøstl
3. JH
4. Luffa
5. Lane

Improvements on Best Known Analysis

Hash Function	SHA3 Round	Best Known Analysis	Rounds / Total	Previous Time	Previous Memory	Ref.	This Paper Time	This Paper Memory
JH	Final	semi-free-start coll.	16 / 42	2^{190}	2^{104}	[RTV10]	$\mathbf{2^{97}}$	$\mathbf{2^{97}}$
JH		semi-free-start near coll.	22 / 42	2^{168}	$2^{143.70}$	[RTV10]	$\mathbf{2^{96}}$	$\mathbf{2^{96}}$
Grøstl-256	Final*	(compr. function property)	10 / 10	2^{192}	2^{64}	[Pey10]	$\mathbf{2^{182}}$	2^{64}
Grøstl-256		(internal permutation dist.)	10 / 10	2^{192}	2^{64}	[Pey10]	$\mathbf{2^{175}}$	2^{64}
Grøstl-512		(compr. function property)	11 / 14	2^{640}	2^{64}	[Pey10]	$\mathbf{2^{630}}$	2^{64}
ECHO-256	2^{nd}	internal permutation dist.	8 / 8	2^{182}	2^{37}	[SLW+10]	$\mathbf{2^{151}}$	2^{67}
Luffa	2^{nd}	semi-free-start coll.	7 / 8	2^{132}	$2^{68.8}$	[KNPRS10]	$\mathbf{2^{112.9}}$ ($\mathbf{2^{104}}$)	$2^{68.8}$ (2^{102})
LANE-256	1^{st}	semi-free-start coll.	6+3 / 6+3	2^{96}	2^{88}	[MNPN+09]	$\mathbf{2^{80}}$	$\mathbf{2^{66}}$
LANE-512		semi-free-start coll.	8+4 / 8+4	2^{224}	2^{128}	[MNPN+09]	2^{224}	$\mathbf{2^{66}}$

Conclusion

- Problem definition that describes the bottleneck of most rebound attacks. Importance of identifying the best situations.

- Several algorithms for solving the problem in different realistic scenarios.

- Applied to previous rebound attacks, improve considerably their complexities, and most important, results useful for future cryptanalysis. So far:

New Applications

- ***Improved Analysis of ECHO-256*** [Jean et al. SAC11], stop-in-the-middle allows the best known compression function results.

- ***Rebound attack on JH42*** [NP et al. Rump Session ECRYPT Hash Workshop11], problem 1 algorithms and correct problem definitions allow for a semi-free-start near-collision for 37 rounds and a permutation distinguisher for the 42 rounds.

- ***Cryptanalysis of ARMADILLO2*** [Abdelraheem et al. eprint11], parallel matching allows cryptanalysis of all the variants.

References

[KNPRS10] D. Khovratovich, M. Naya-Plasencia, A. Röck, and M. Schläffer. Cryptanalysis of Luffa v2 components. In *SAC*, volume 6544 of *Lecture Notes in Computer Science*, pages 388–409, 2010.

[MNPN+09] Krystian Matusiewicz, María Naya-Plasencia, Ivica Nikolic, Yu Sasaki, and Martin Schläffer. Rebound Attack on the Full LANE Compression Function. In *ASIACRYPT*, volume 5912 of *Lecture Notes in Computer Science*, pages 106–125. Springer, 2009.

[Pey10] Thomas Peyrin. Improved Differential Attacks for ECHO and Grøstl. In *Advances in Cryptology - CRYPTO 2010, 30th Annual Cryptology Conference, Santa Barbara, CA, USA, August 15-19, 2010. Proceedings*, volume 6223 of *Lecture Notes in Computer Science*, pages 370–392. Springer, 2010.

[RTV10] Vincent Rijmen, Denis Toz, and Kerem Varici. Rebound Attack on Reduced-Round Versions of JH. In *FSE*, volume 6147 of *Lecture Notes in Computer Science*, pages 286–303, 2010.

[SLW+10] Y. Sasaki, Y. Li, L. Wang, K. Sakiyama, and K. Ohta. Non-Full-Active Super-Sbox Analysis Applications to ECHO and Grøstl. In *ASIACRYPT*, volume 6477 of *Lecture Notes in Computer Science*, pages 38–55, 2010. To appear.