Time-Space Tradeoffs for Bounded-Length collisions in Merkle-Damgård hashing

Ashrujit Ghoshal
University of Washington

Ilan Komargodski
Hebrew University and NTT Research

CRYPTO 2022

Iterative hashing

Hash functions need to handle variable input lengths

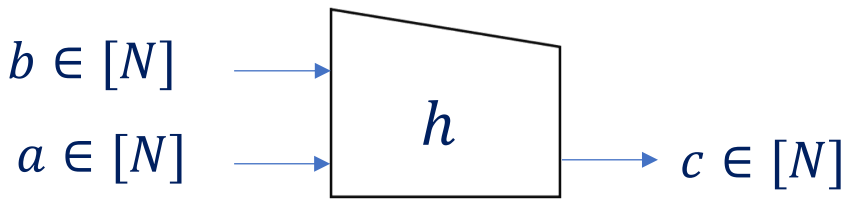
- password hashing
- hash and sign
- commitments

Cannot design a different hash for every length

Construct a VIL hash function from an underlying FIL primitive

e.g., Merkle Damgård hashing [Mer89, Dam89], sponge [BDPV07]

Merkle-Damgård

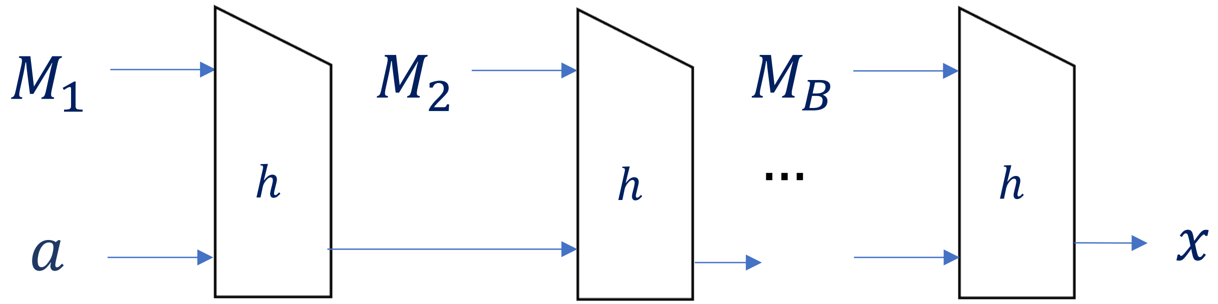


$\mathrm{MD}_h(a, M) = x$

$M = (M_1, M_2, \dots, M_B)$

salt

$\in [N]$



Used in MD5, SHA-1, SHA-2

Collision resistance:

Given a random salt a, hard to find $M \neq M'$ such that $\mathrm{MD}_h(a, M) = \mathrm{MD}_h(a, M')$

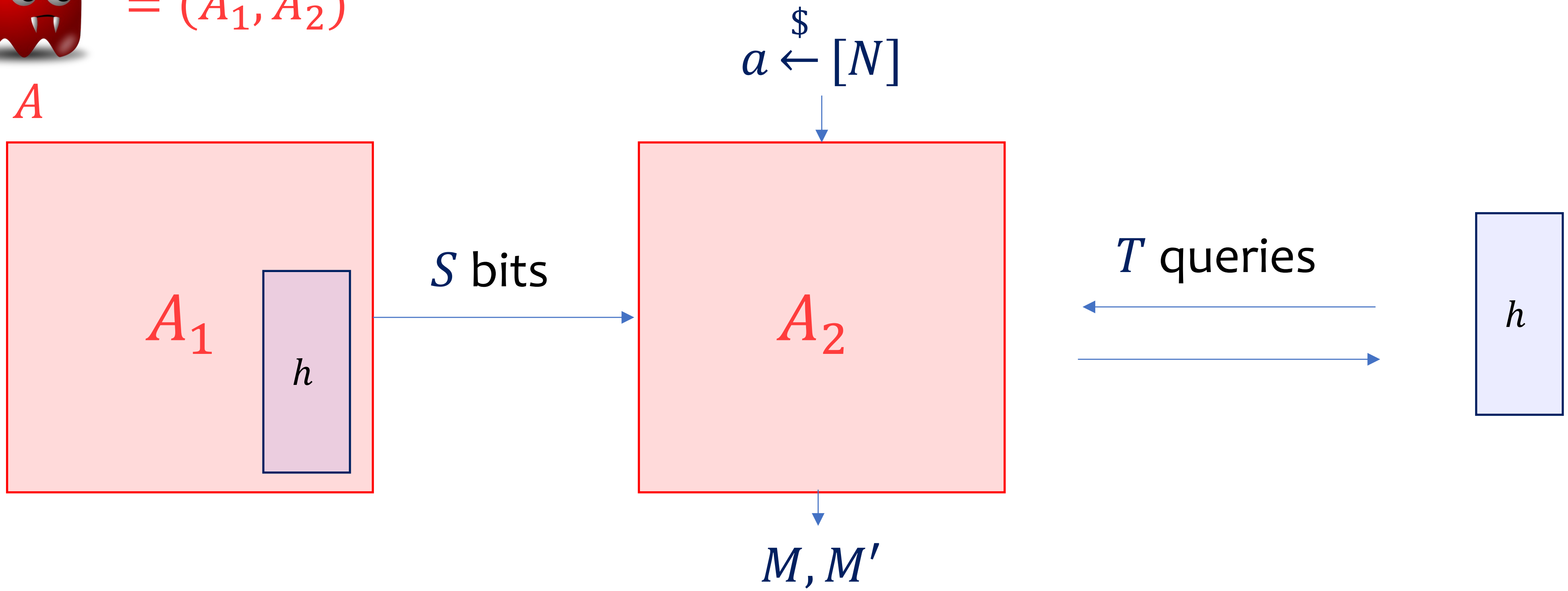
Complexity of finding collisions

- Model h as a random oracle
- Using $T \approx \sqrt{N}$ queries, can find collisions
  - This is necessary
- What about adversaries with large preprocessing?
  - birthday-style attack no longer optimal
  - Scenario studied by [Hellman80, Fiat-Naor99, Unruh07,…]

Auxiliary-input random oracle model (AI-ROM) [Unruh07]

$= (A_1, A_2)$

A



“pre-processing” phase

“online” phase

A wins if $M' \neq M$, $\mathrm{MD}_h(a, M) = \mathrm{MD}_h(a, M')$

$$\mathrm{Adv}_N(S,T) = \max_{(S,T)\text{ adv } A} \Pr[A \text{ wins}]$$

Prior work

Theorem. [CDGS18] $\mathrm{Adv}_N(S,T) = \Theta\left(\frac{ST^2}{N}\right)$

An observation: the attack finds collisions of length $\Omega(T)$!

Say, $T \approx 2^{60} \Rightarrow$ petabytes sized collision!

2-block collision

Shorter collisions are provably harder to find

Theorem. [ACDW20] $\mathrm{Adv}_{N,2}(S,T) \le O\left(\frac{ST}{N} + \frac{T^2}{N}\right)$

Theorem (STB attack). [ACDW20] $\mathrm{Adv}_{N,\boldsymbol{B}}(S,T) \geq \widetilde{\Omega}\left(\frac{STB}{N} + \frac{T^2}{N}\right)$

The STB conjecture [ACDW20]

"the optimal attack for finding B-block collisions has advantage at most $\tilde{O}\left(\frac{STB}{N} + \frac{T^2}{N}\right)$"

Was unresolved for $\boldsymbol{3 \leq B \ll T}$

This work:

Proof of the STB conjecture for

- $B = O(1)$ **Next**
- $S^4 B^2 \in \tilde{O}(T)$ **See paper**

Recently improved by Akshima, Guo, Liu [AGL22]

Main theorem

Theorem. [this work]

$$\mathrm{Adv}_{N,B}(S,T) \leq O\left(\frac{STB^2(\log S)^B}{N} + \frac{T^2}{N}\right)$$

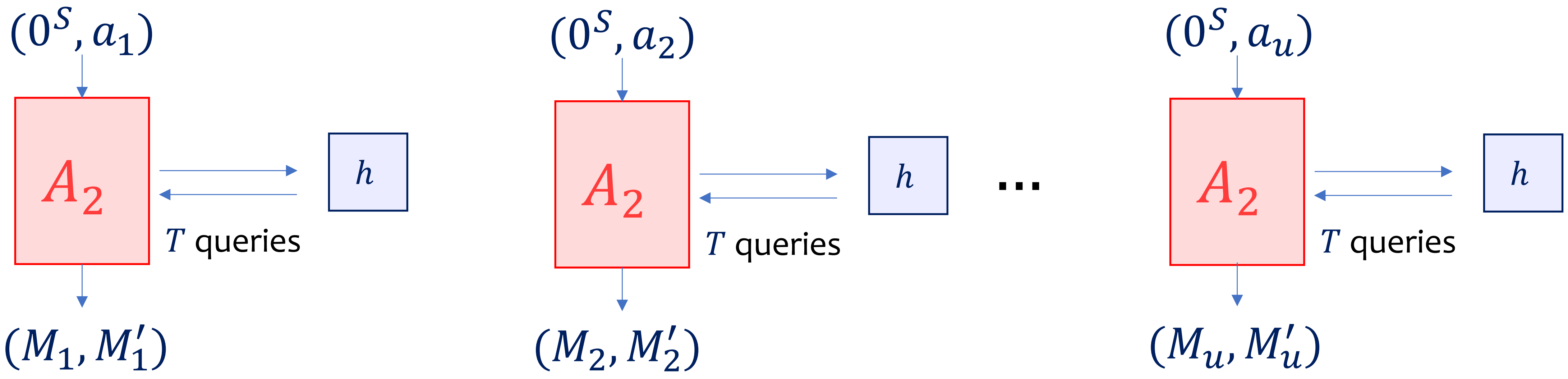
For constant B,

$$\mathrm{Adv}_{N,B}(S,T) \leq \tilde{O}\left(\frac{ST}{N} + \frac{T^2}{N}\right)$$

Proof via multi-instance framework [IK10, CGLQ20, ACDW20]

Multi-instance framework [CGLQ20, ACDW20]

$$a_1, a_2, \ldots, a_u \xleftarrow{\$} [N]$$



A_2 wins if $\forall i \in [u]$

1. $M_i \neq M_i'$ 2. $\mathrm{MD}_h(a_i, M_i) = \mathrm{MD}_h(a_i, M_i')$ 3. $|M_i|, |M_i'| \leq B$

Multi-instance lemma. Let $u = S + \log N$. Define $\varepsilon := \max_{A_2} \Pr[A_2 \text{ wins}]$. Then

$$\mathrm{Adv}_{N,B}(S,T) \le \varepsilon^{\frac{1}{u}}$$

Will prove:

$$\varepsilon \le \left(O\left(\frac{uTB^2(\log u)^B}{N} + \frac{T^2}{N} \right) \right)^u$$

For constant B, $u = S + \log N$

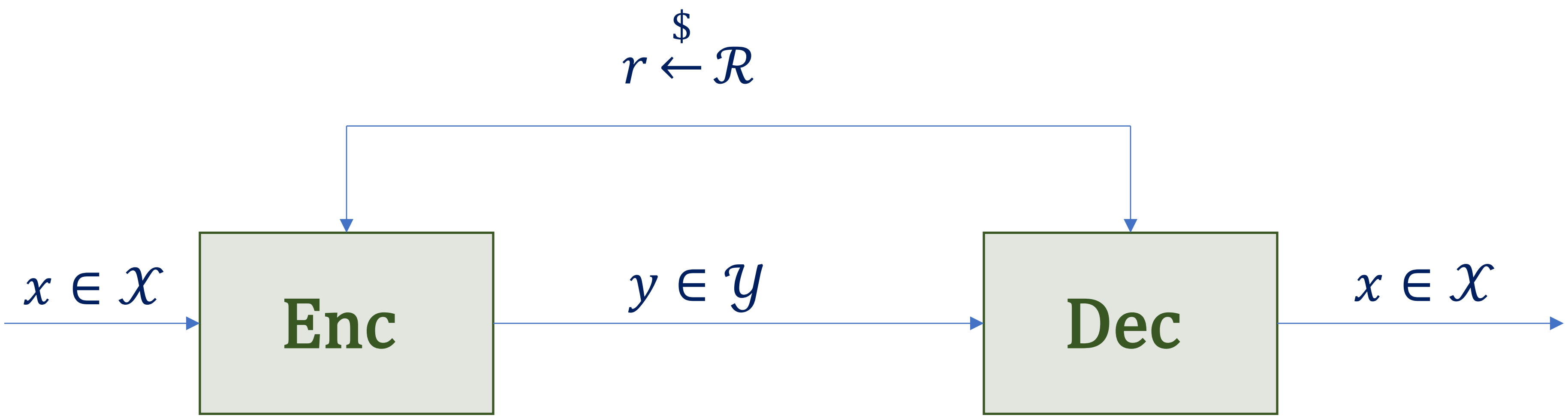
$$\varepsilon \le \left(\tilde{\boldsymbol{O}}\left(\frac{ST}{N} + \frac{T^2}{N} \right) \right)^u$$

From multi-instance lemma, it follows

$$\mathrm{Adv}_{N,B}(S,T) \le \tilde{\boldsymbol{O}}\left(\frac{ST}{N} + \frac{T^2}{N} \right)$$ ✓

Upper bounding multi-instance advantage

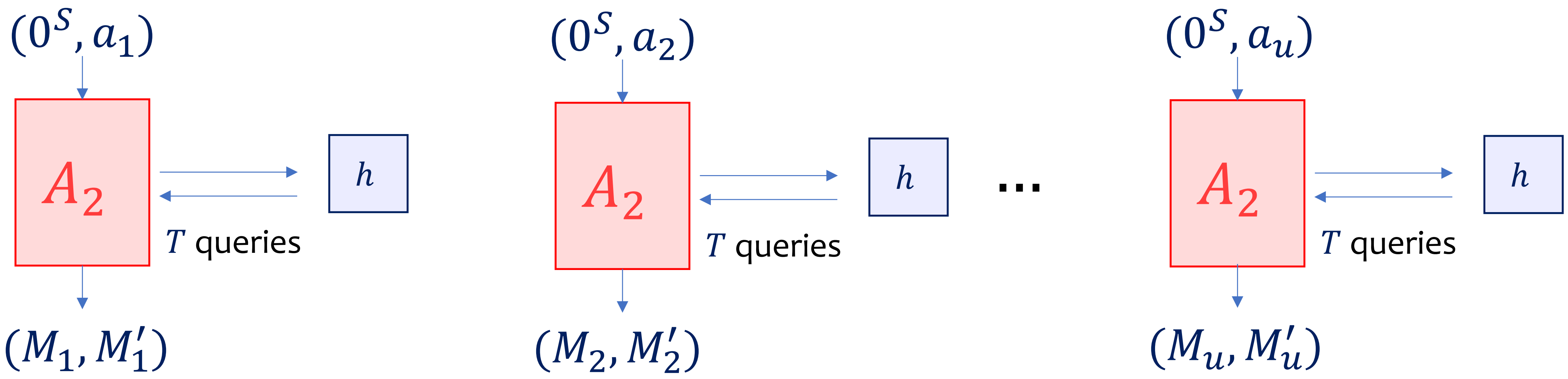
Technique: compression argument



Lemma [GT00,DTT10]. Let $\varepsilon := \Pr_{x,r}[\mathrm{Dec}(\mathrm{Enc}(x,r),r) = x]$. Then

$$\log|\mathcal{Y}| \geq \log|\mathcal{X}| - \log\frac{1}{\varepsilon}$$

$$a_1, a_2, \ldots, a_u \xleftarrow{\$} [N]$$

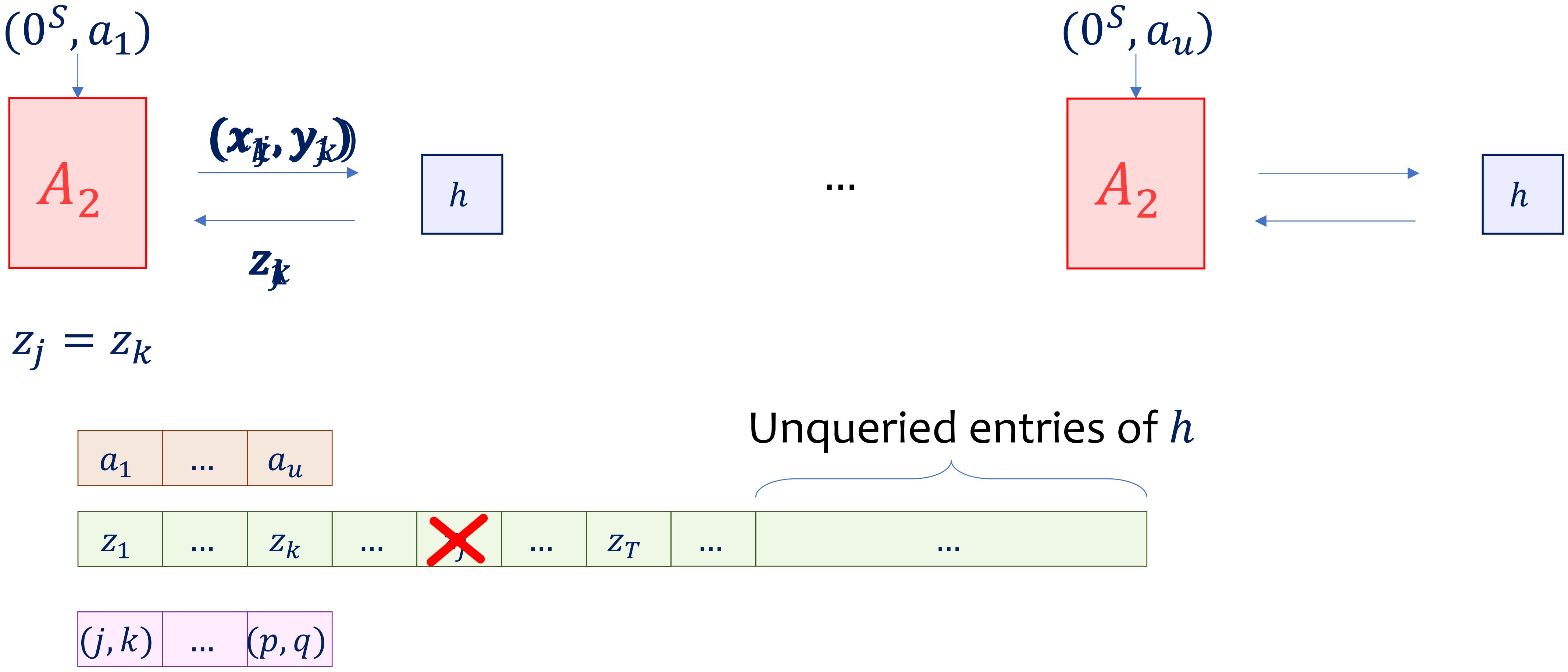


Our strategy: Encode $h, \{a_1, a_2, \ldots, a_u\}$ using A_2 that always wins

Compression lemma $\Rightarrow$ upper bound $\Pr[A_2 \text{ wins}]$

Simplifying assumption: Only queries of the form $h(a_i, *)$ when A_2 run on a_i

Encoding



$(0^S, a_1)$
A_2
(x_j, y_j)
(x_k, y_k)
h
z_j
z_k
…
$(0^S, a_u)$
A_2
h
$z_j = z_k$
Unqueried entries of h
a_1 | … | a_u
z_1 | … | z_k | … | z_j | … | z_T | … | …
(j, k) | … | (p, q)

Decoding

Collision for every salt

Unqueried entries of h

$\Rightarrow$ Savings $= u(\log N - 2\log T)$

omit an answer

remember two indices

a_1	…	a_u

Using the compression lemma,

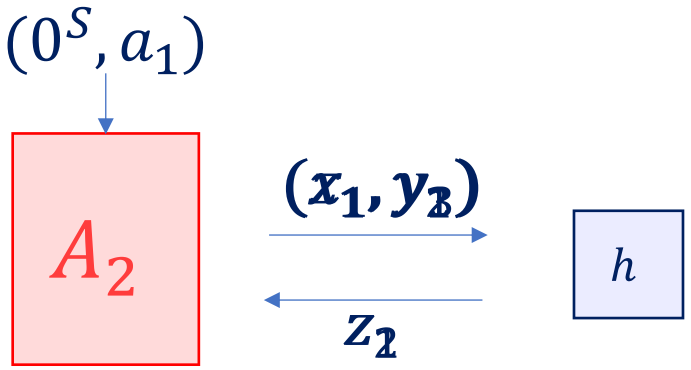
$$\varepsilon \le \left(\frac{T^2}{N}\right)^u$$

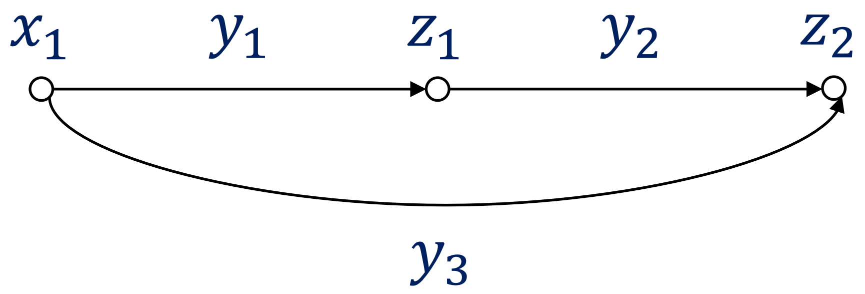
z_1	…	z_k	…	$\cancel{z_j}$	…	z_T	…	…

(j,k)	…	(p,q)

However, cannot assume only queries of the form $h(a_i,*)$ are made when A_2 run on a_i

Query graph





Graph grows across all of A_2's runs

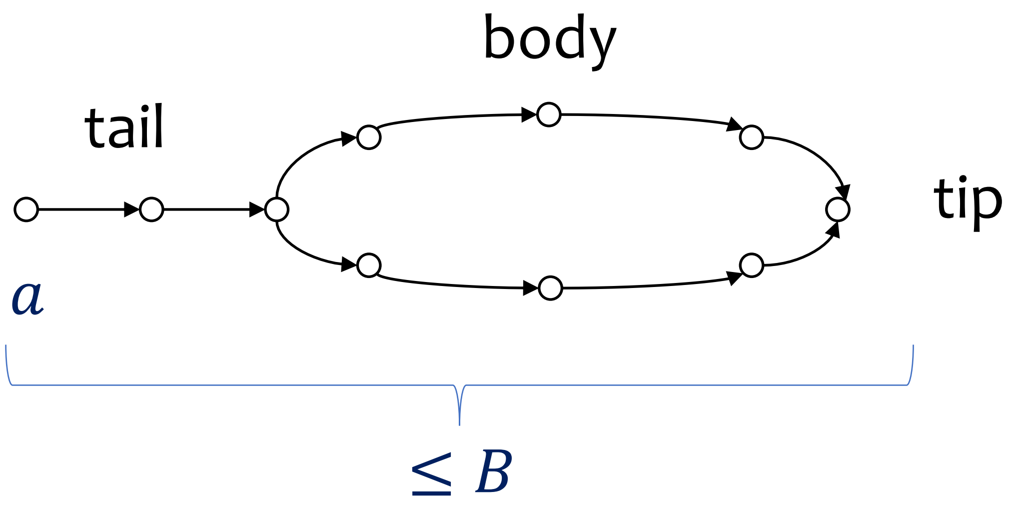
Note: A_2 may repeat queries across different runs

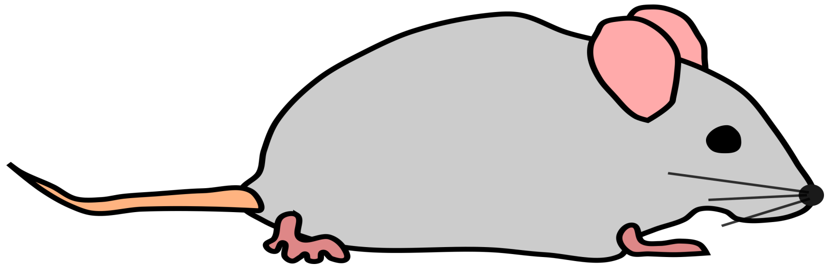
Assume wlog A_2 makes all h queries needed to compute collision

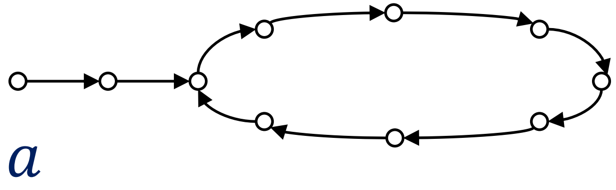
How do B-block collisions look like?

Collision structure

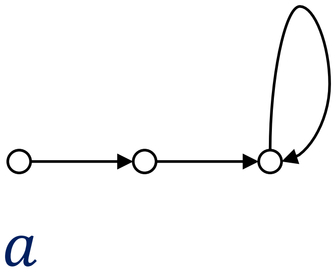
The mouse structure



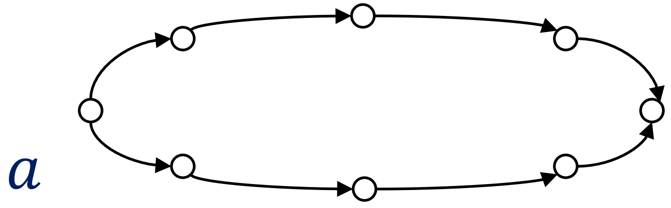




no lower body



Self loop body



No tail

Isolate **one** mouse structure per salt

Types of queries

- **New** queries: queries made for the first time
  - wlog no queries repeated in single A_2 run
  - query not made in any previous A_2 run $\Rightarrow$ new query

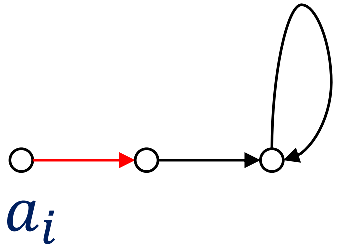
- Repeated queries
  - **repeated-mouse** queries: query present in some earlier mouse structure
  - **repeated-non-mouse** queries: other queries

Assume: Before running A_2 on a_i, $h(a_i,*)$ not queried

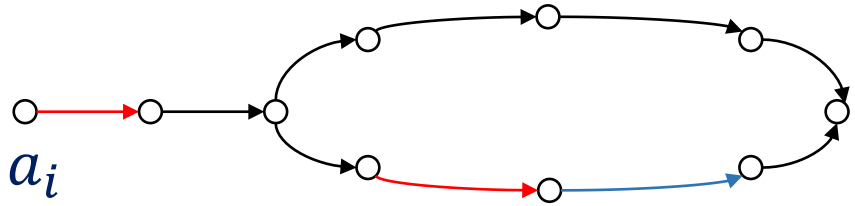
$\Rightarrow$ every mouse structure has a new query

Classifying mouse structures

1) Colliding new queries

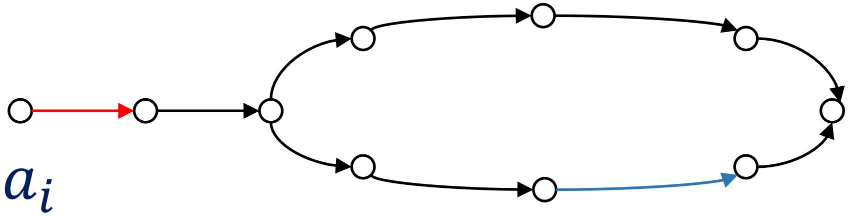


2) Self loop body

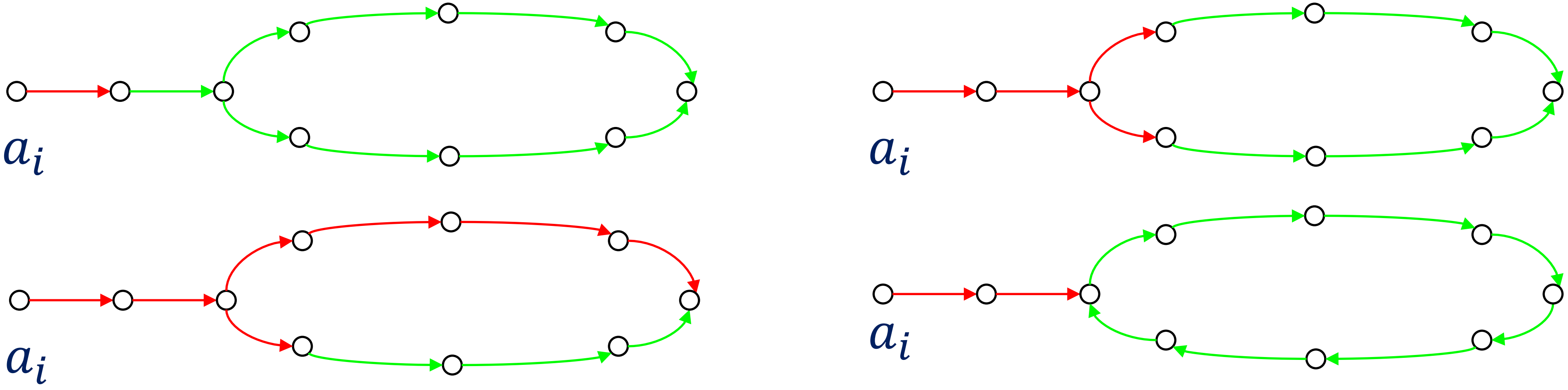


3) New query touching repeated-mouse query

Classifying mouse structures (2)



4. At least one repeated-mouse query



5. No repeated-mouse query

Goal: for every mouse structure save at least

$$\delta = \min\left\{\log\frac{N}{T^2}, \log\frac{N}{uTB^2(\log u)^B}\right\} \quad \text{bits}$$

Total savings $\geq\ u \cdot \delta$ bits

Using the compression lemma,

$$\varepsilon \leq \max\left\{\frac{T^2}{N}, \frac{4uTB^2(3\log u)^B}{N}\right\} \leq \left(O\left(\frac{uTB^2(\log u)^B}{N} + \frac{T^2}{N}\right)\right)^u \quad ✓$$

Recall assumption: Before running A_2 on a_i, $h(a_i,*)$ not queried

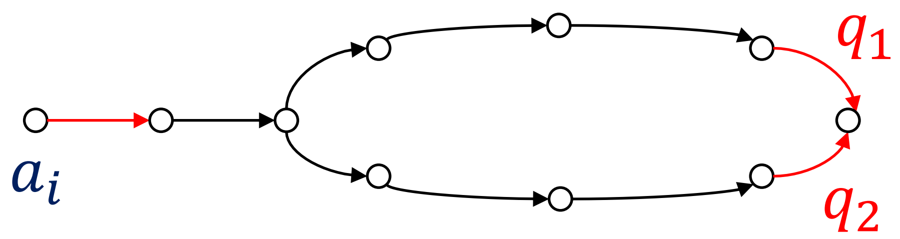
Why is it reasonable?

Because otherwise save on a_i

$$\text{Savings} = \underbrace{\log N}_{\text{omit } a_i} - \underbrace{\log uT}_{\text{add query index of } h(a_i,*)} \geq \delta$$

That suffices!

Easy case examples

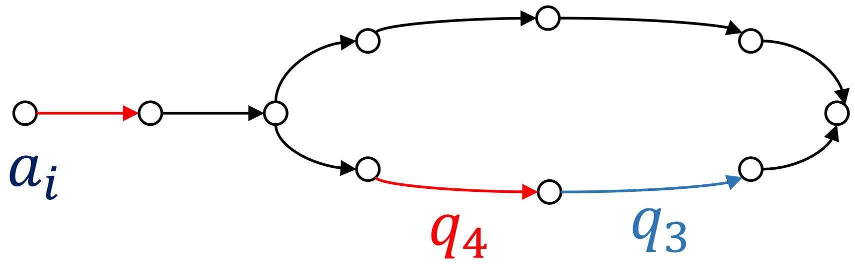


Colliding new queries

Say q_2 after q_1

Savings:

$$\underbrace{\log N}_{\text{answer of } q_2} - \underbrace{2\log T}_{\text{“local” indices of } q_1, q_2} \geq \delta$$

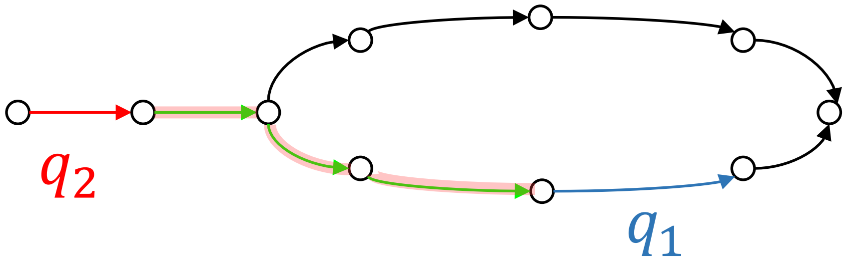


Savings:

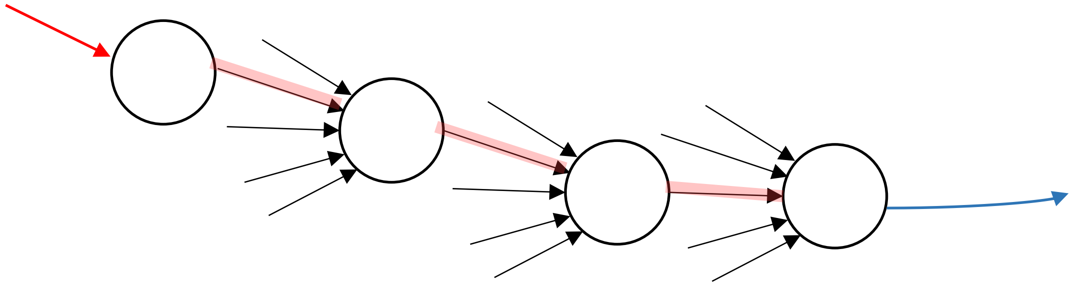
$$\underbrace{\log N}_{\text{answer of } q_4} - \underbrace{\log T}_{\text{index of } q_4} - \underbrace{\log uB}_{\text{index of } q_3} \geq \delta$$

New query touching repeated-mouse query

Hard case example



At least one repeated-mouse query



Strategy:
Omit answer of q_2,
Remember:

- index of q_1
- index of q_2
- **path** back from q_1 to q_2

No large multi-collision if:
$\leq \log u$ incoming edges for all nodes

no large multi-collision $\Rightarrow$ **path** encoding needs at most

$$\underbrace{\log B}_{\text{\# of edges on path}} + \underbrace{B \log(\log u)}_{\text{which edge to take on path back}}$$

Strategy:
Omit answer of q_2,
Remember:

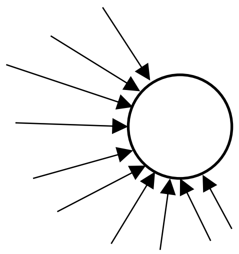
- index of q_1
- index of q_2
- **path** back from q_1 to q_2

Savings $\geq \log \mathrm{N} - (\log uB + \log T + \log(\log u)^B + \log B) \geq \delta$

But, what if there are large muti-collisions?

Key idea: Save from the large multi-collision!

Saving from multi-collisions



m- multi-collision

Strategy:
Remember answer of first of m queries, indices of rest

Savings:

$$\underbrace{(m-1)\log N}_{\text{omitted answers}} - \underbrace{\log\binom{uT}{m}}_{\text{set of query indices}}$$

When $m \geq \log u$,

$$(m-1)\log N - \log\binom{uT}{m} \geq \log N - 2\log T \geq \delta$$

Conclusion

- STB conjecture true for all constant B, when $S^4B^2 \in \tilde{O}(T)$
- Follow up works
  - STB conjecture proven for $ST^2 \leq N$ [AGL22]
  - similar question studied for sponge [FGK22]

Open problem:
Prove the STB conjecture or give better attacks for $ST^2 > N$

Paper: https://eprint.iacr.org/2022/309