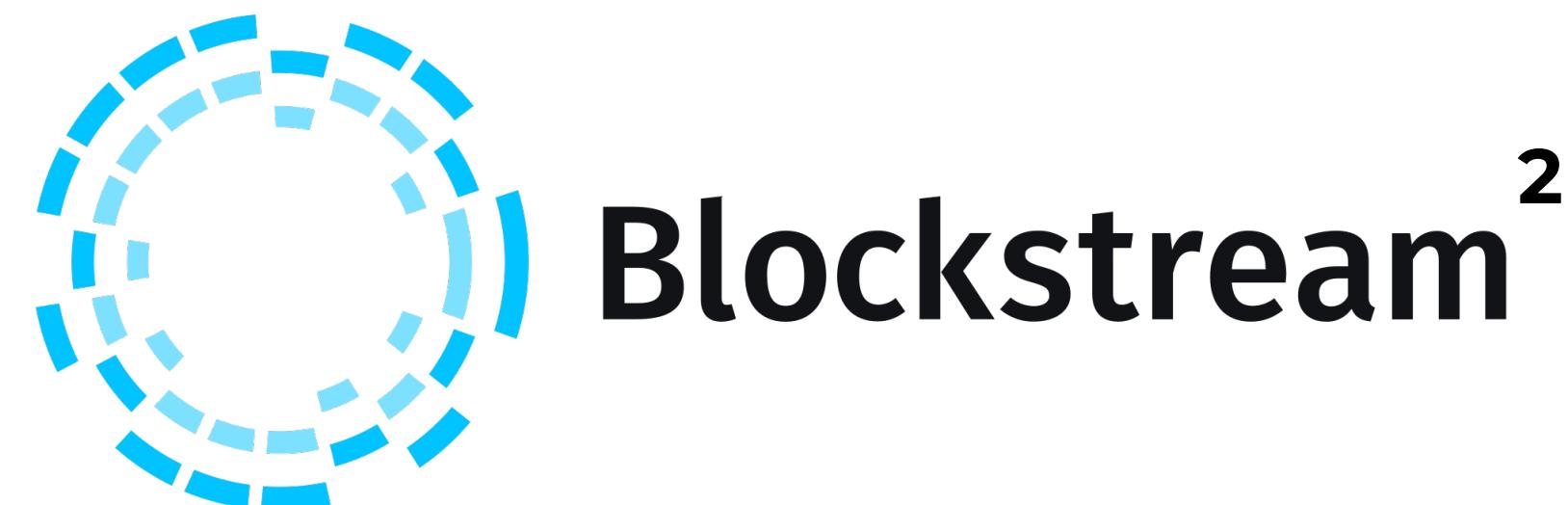
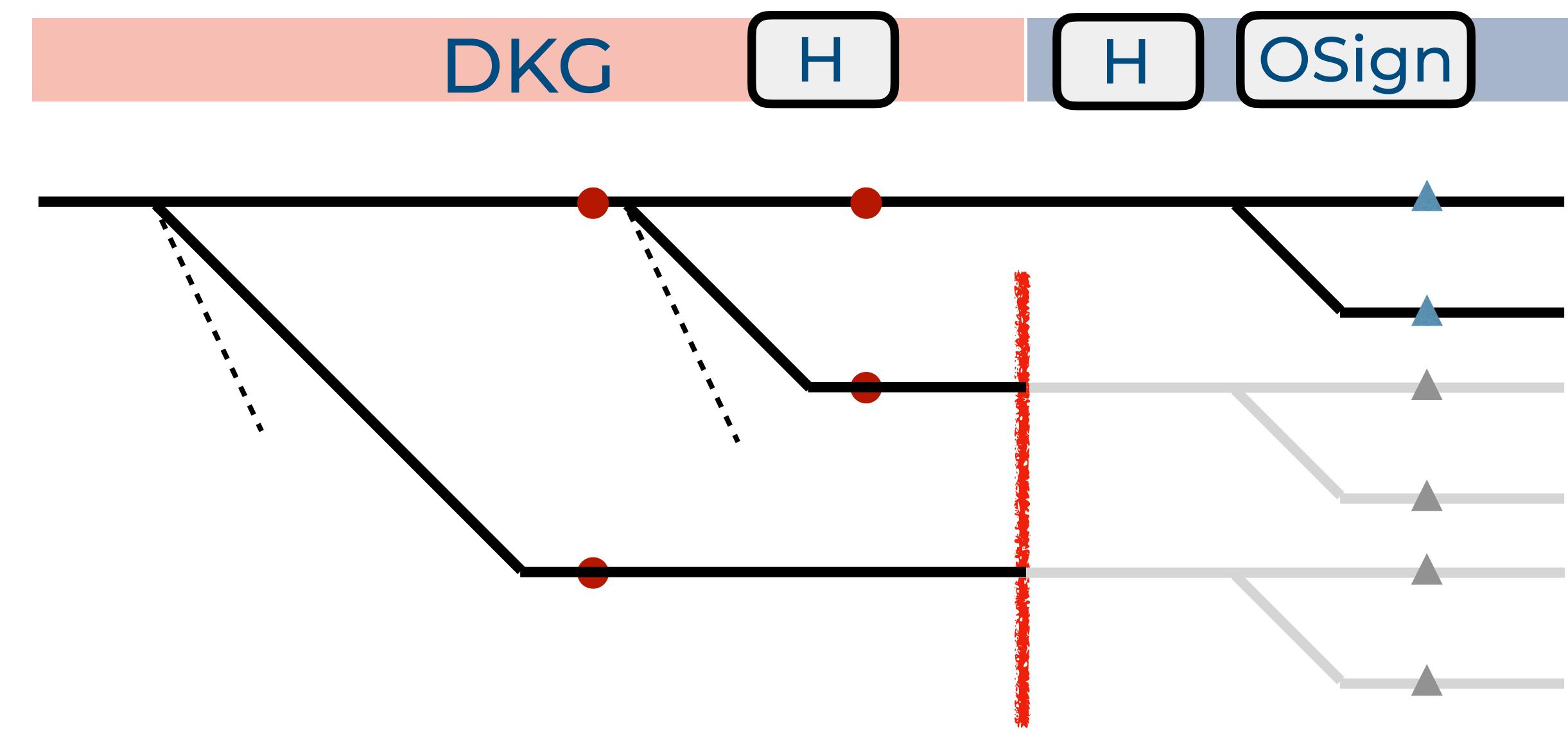
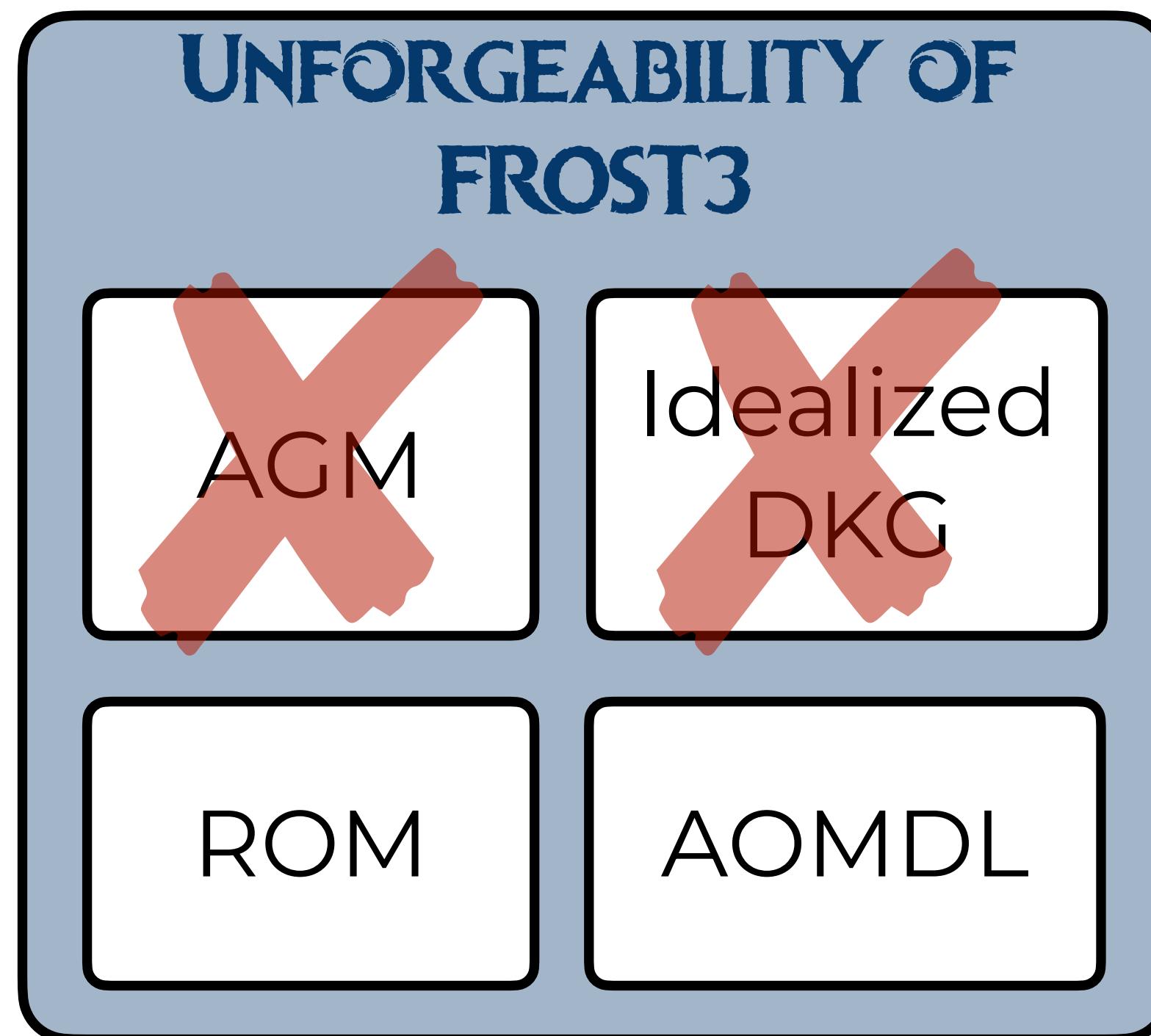


Practical Schnorr Threshold Signatures Without the Algebraic Group Model

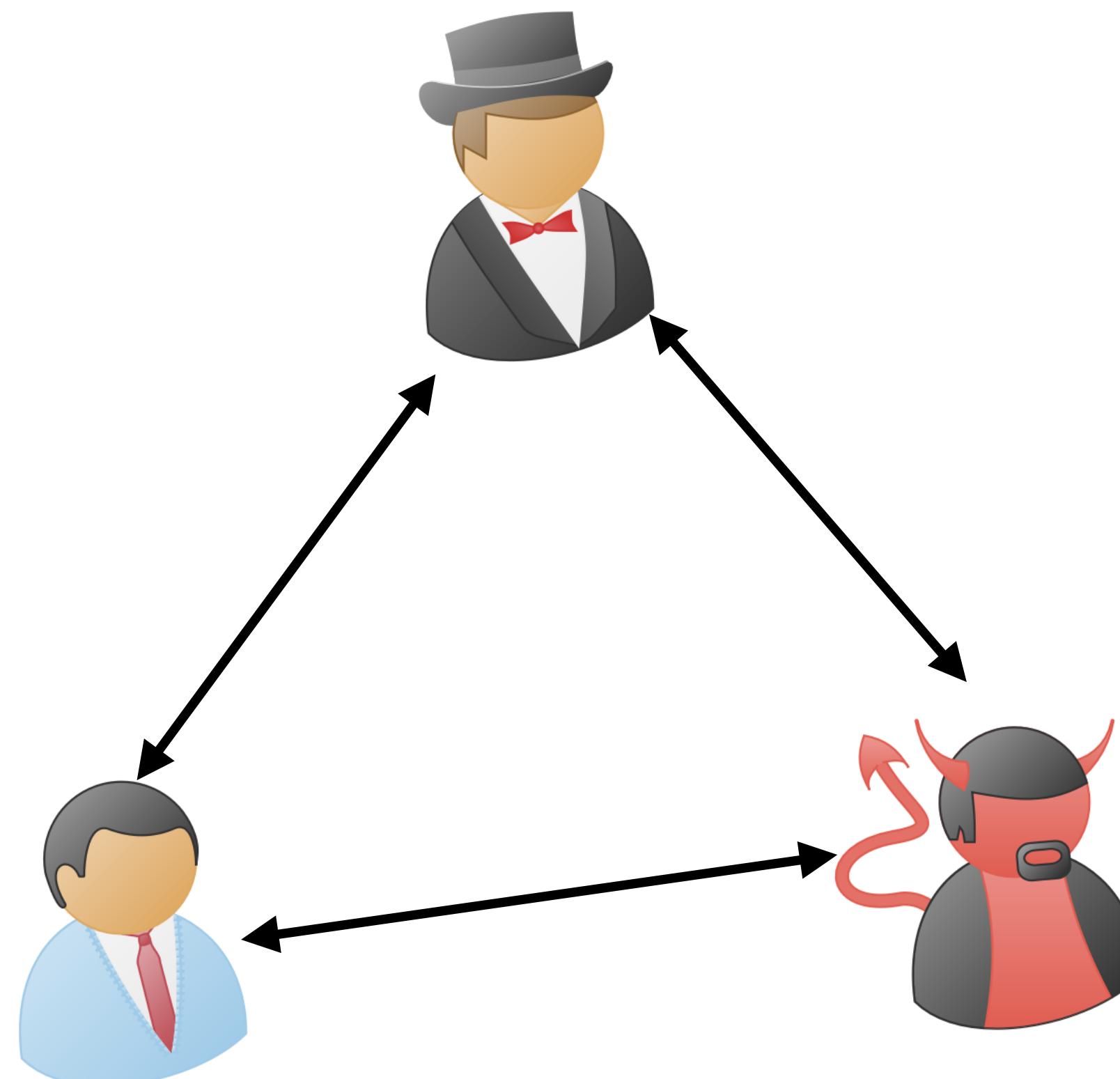
Hien Chu¹, Paul Gerhart¹, Tim Ruffing², and Dominique Schröder¹



Our Contributions



Threshold Signatures [DES88, DF90]



$m = \text{LET'S PROVE FROST3}$

$$(\sigma_1, \sigma_2, \sigma_3) \xrightarrow{\text{Aggregate}} \sigma = (R, s)$$

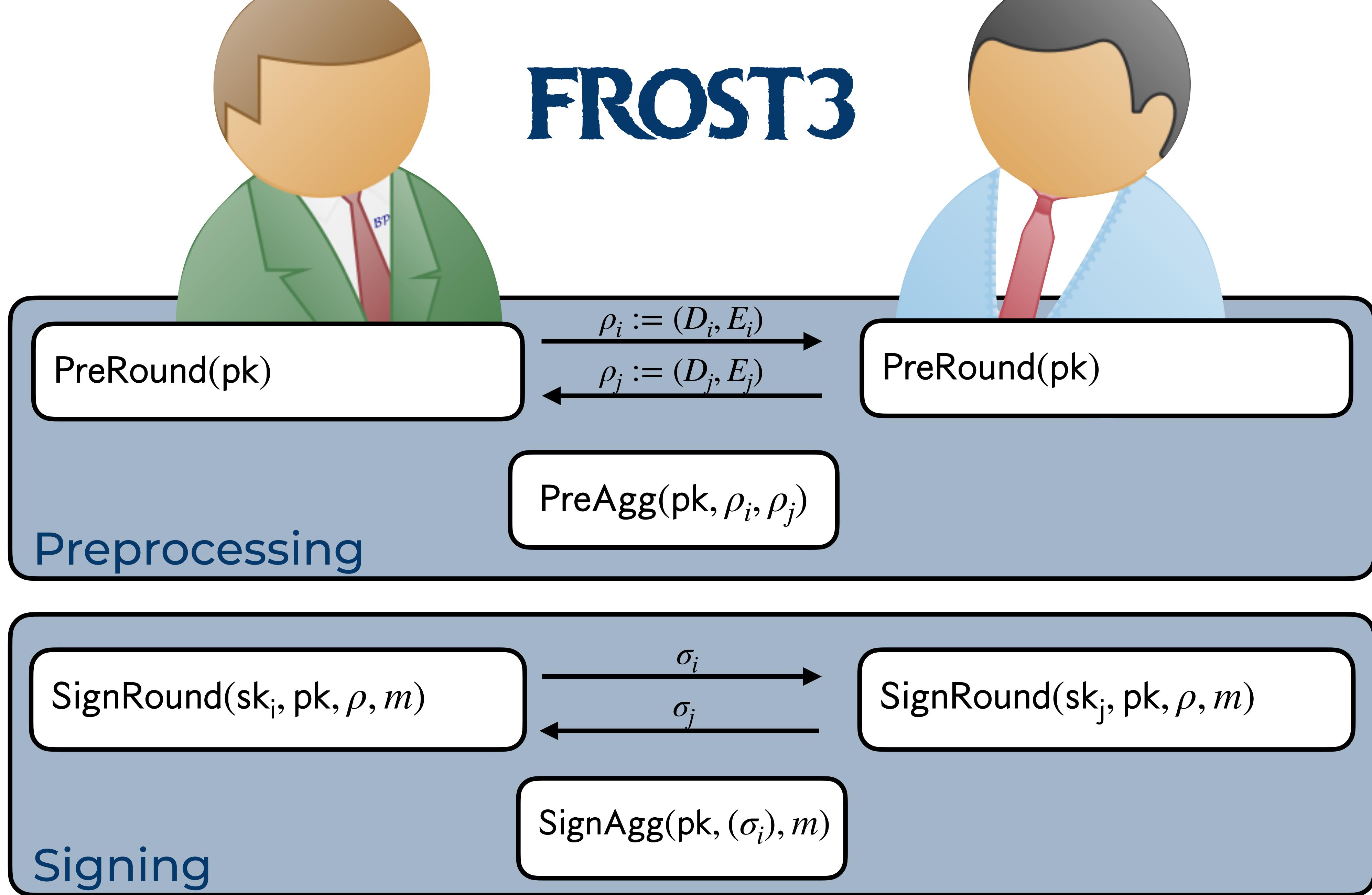
$$c := H(\text{pk}, R, m)$$

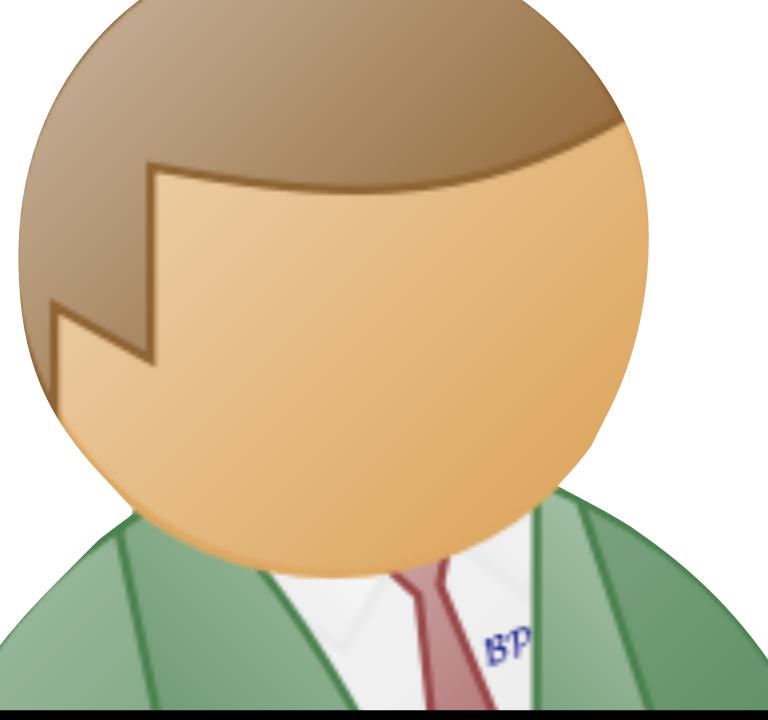
$$R \stackrel{?}{=} g^s \text{pk}^{-c}$$

FROST

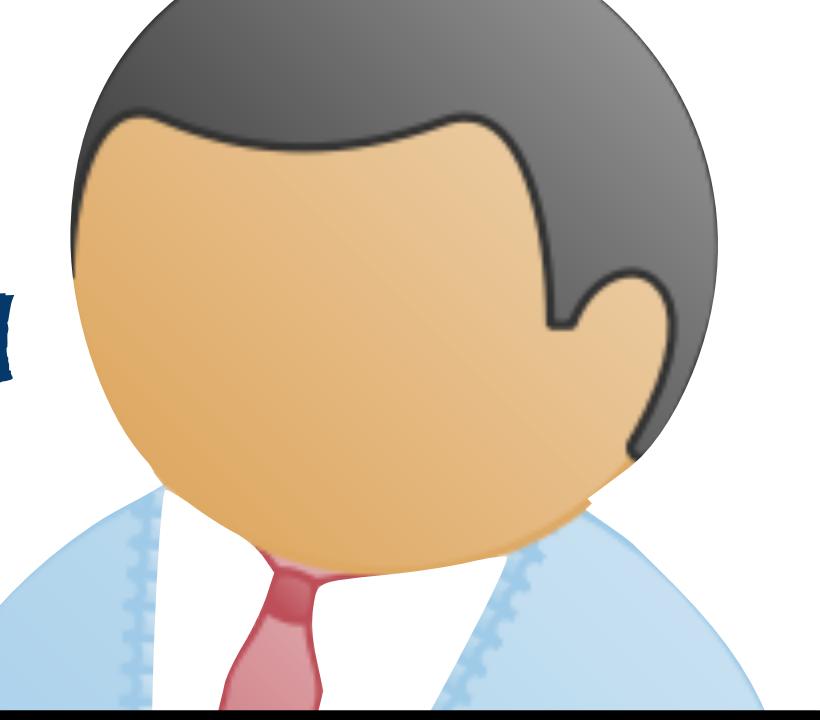
Komlo and Goldberg
SAC 2020

FROST3

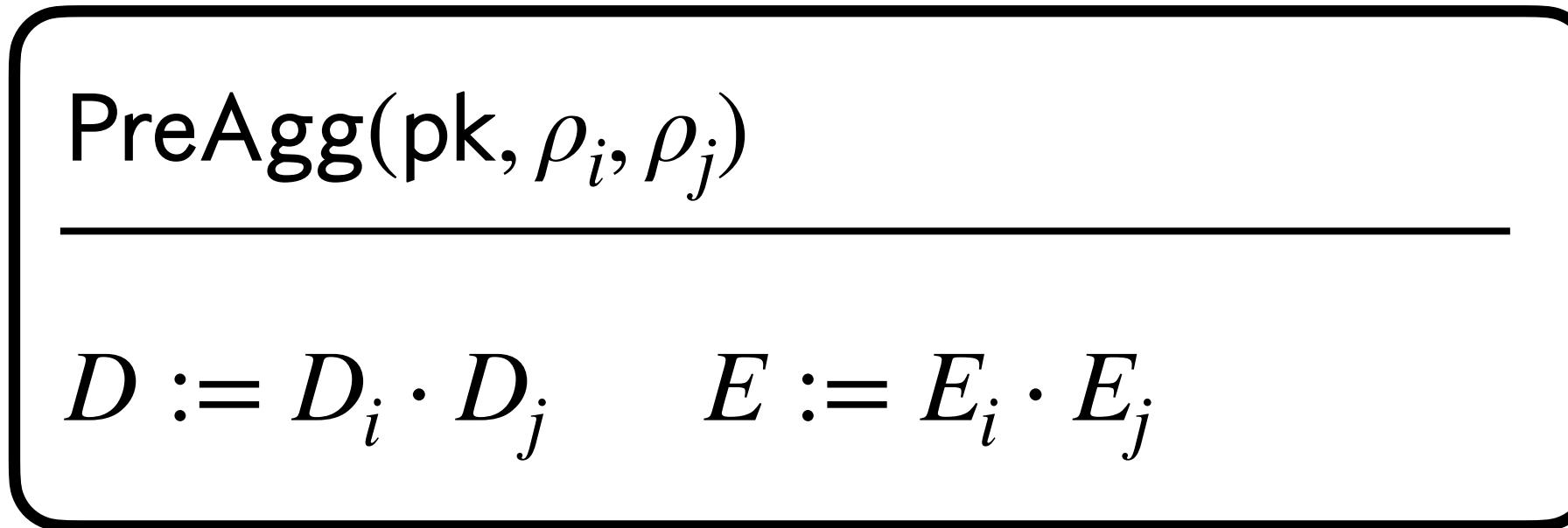
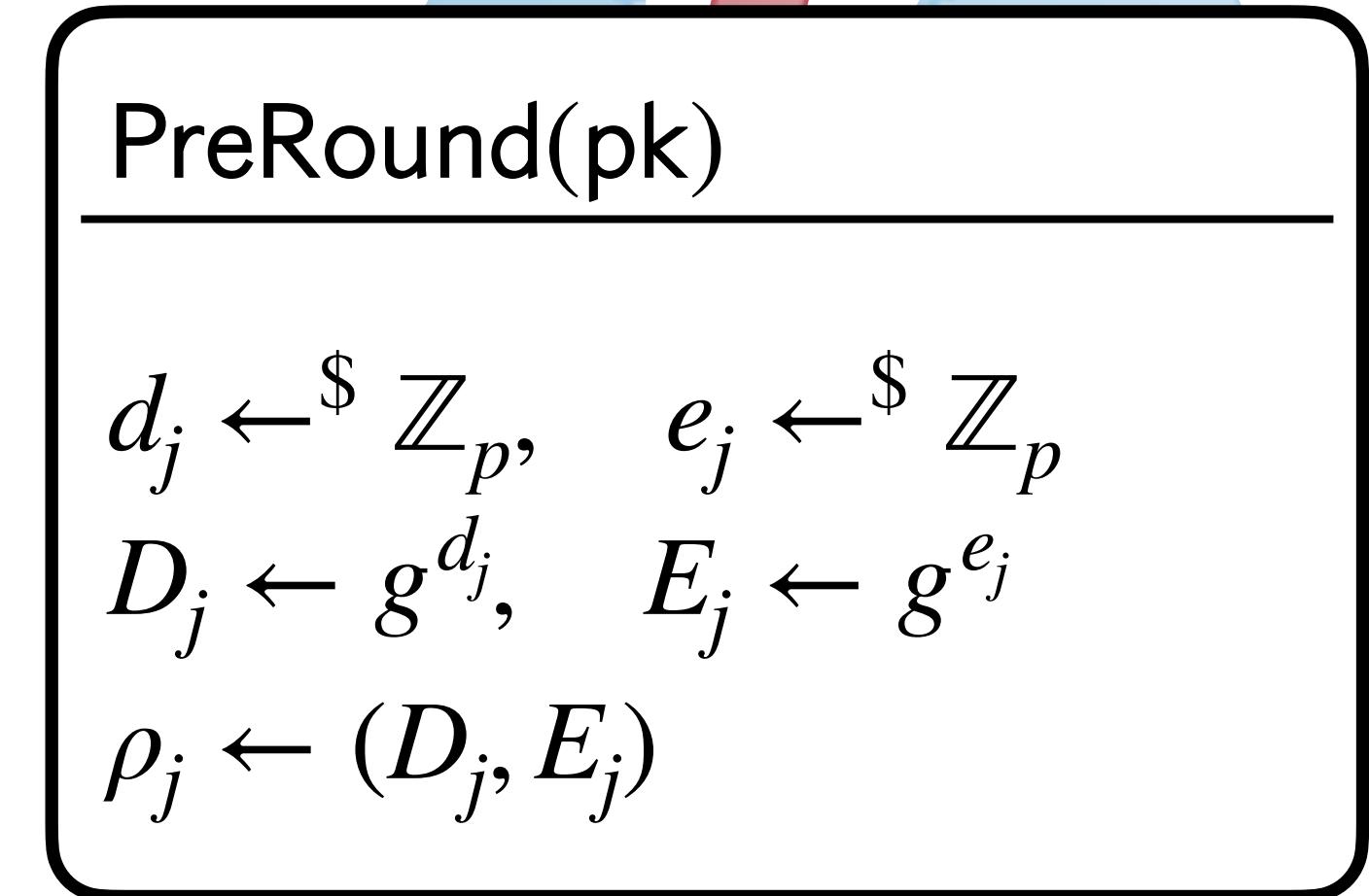
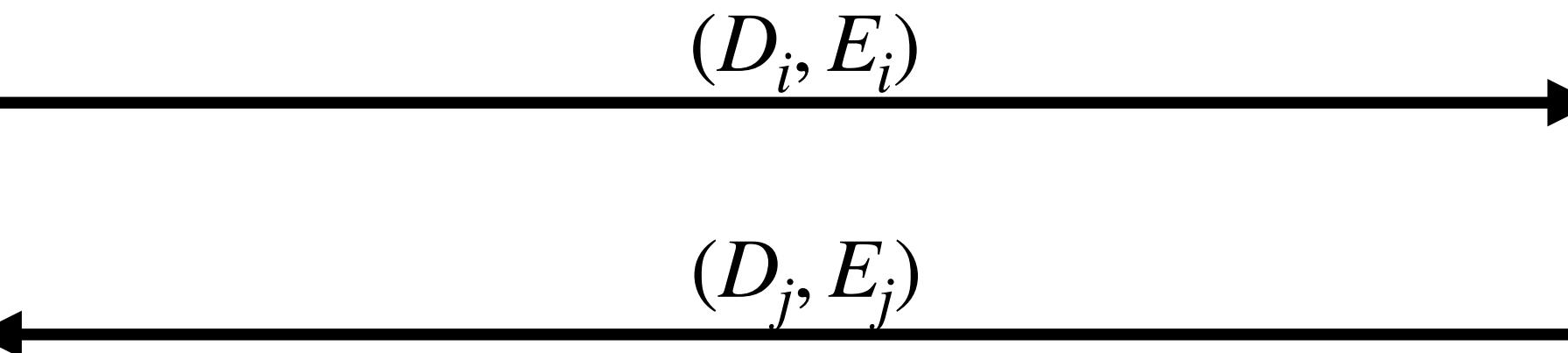
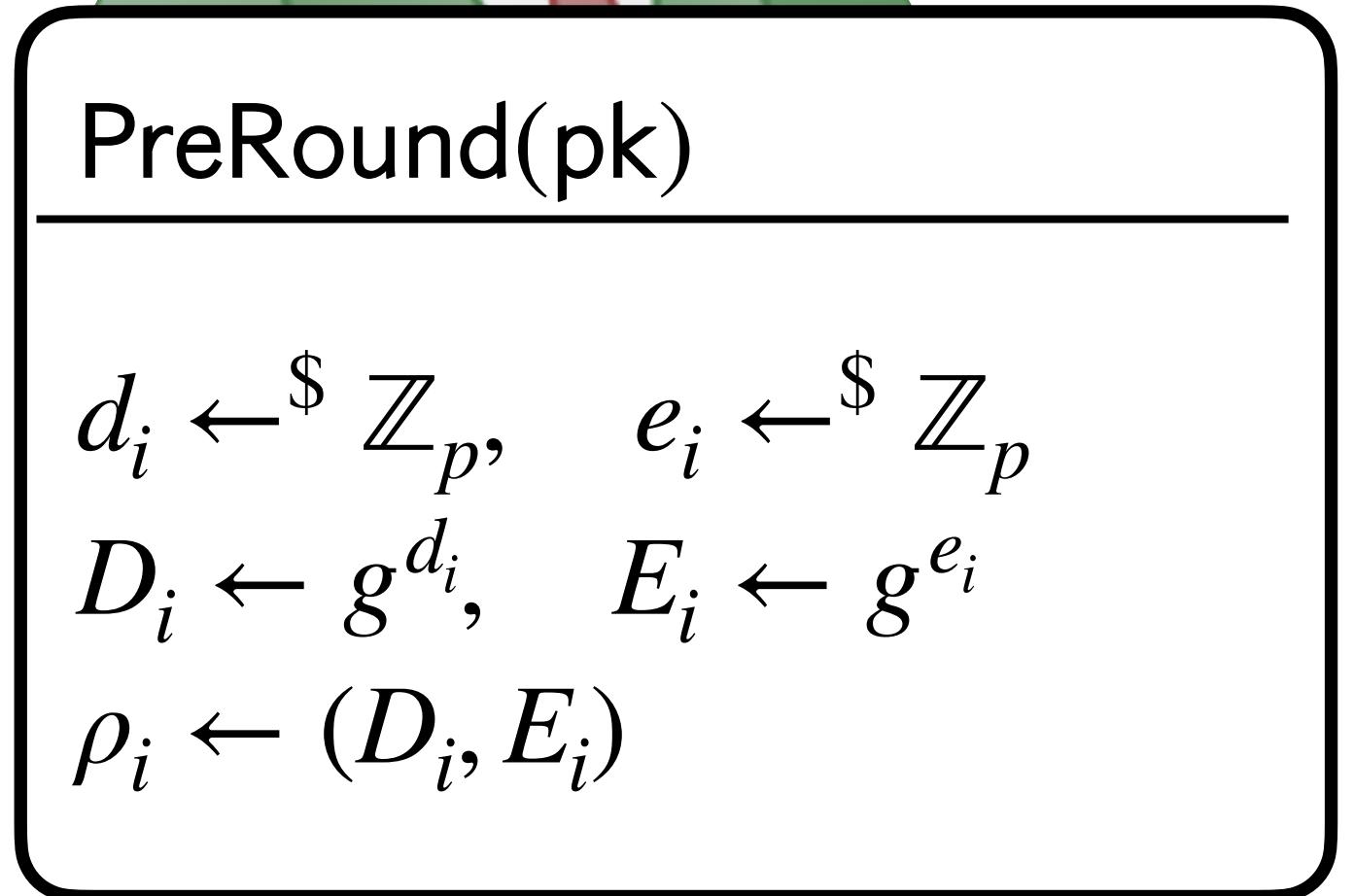




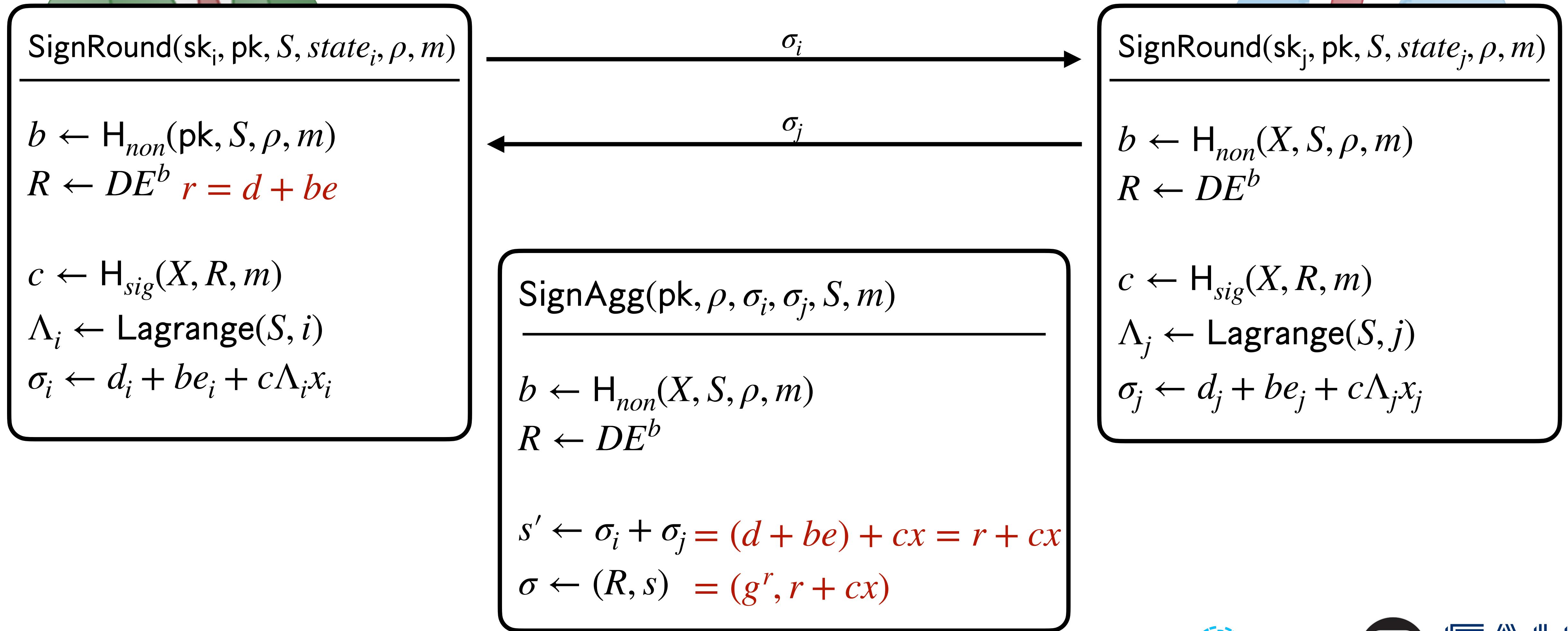
FROST3 PREPROCESSING



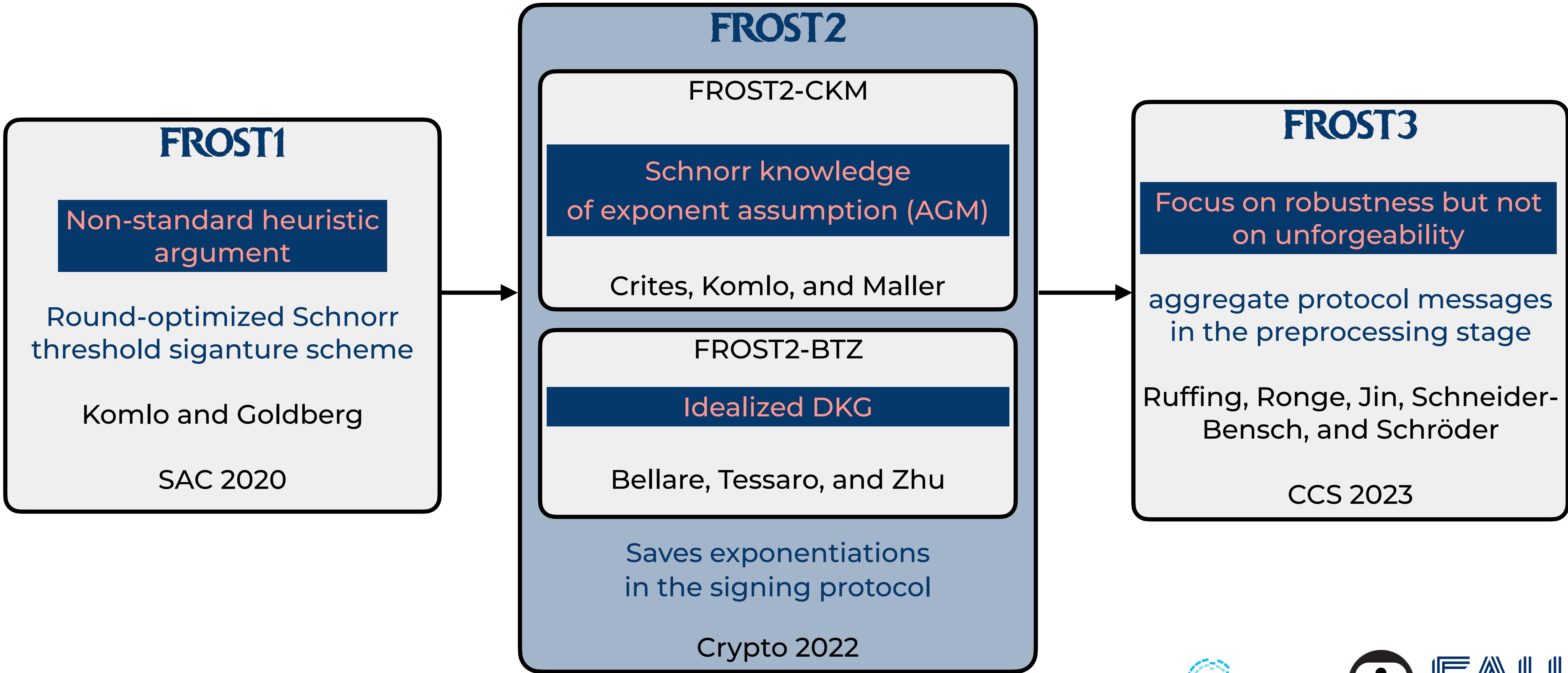
(SIMPLIFIED)



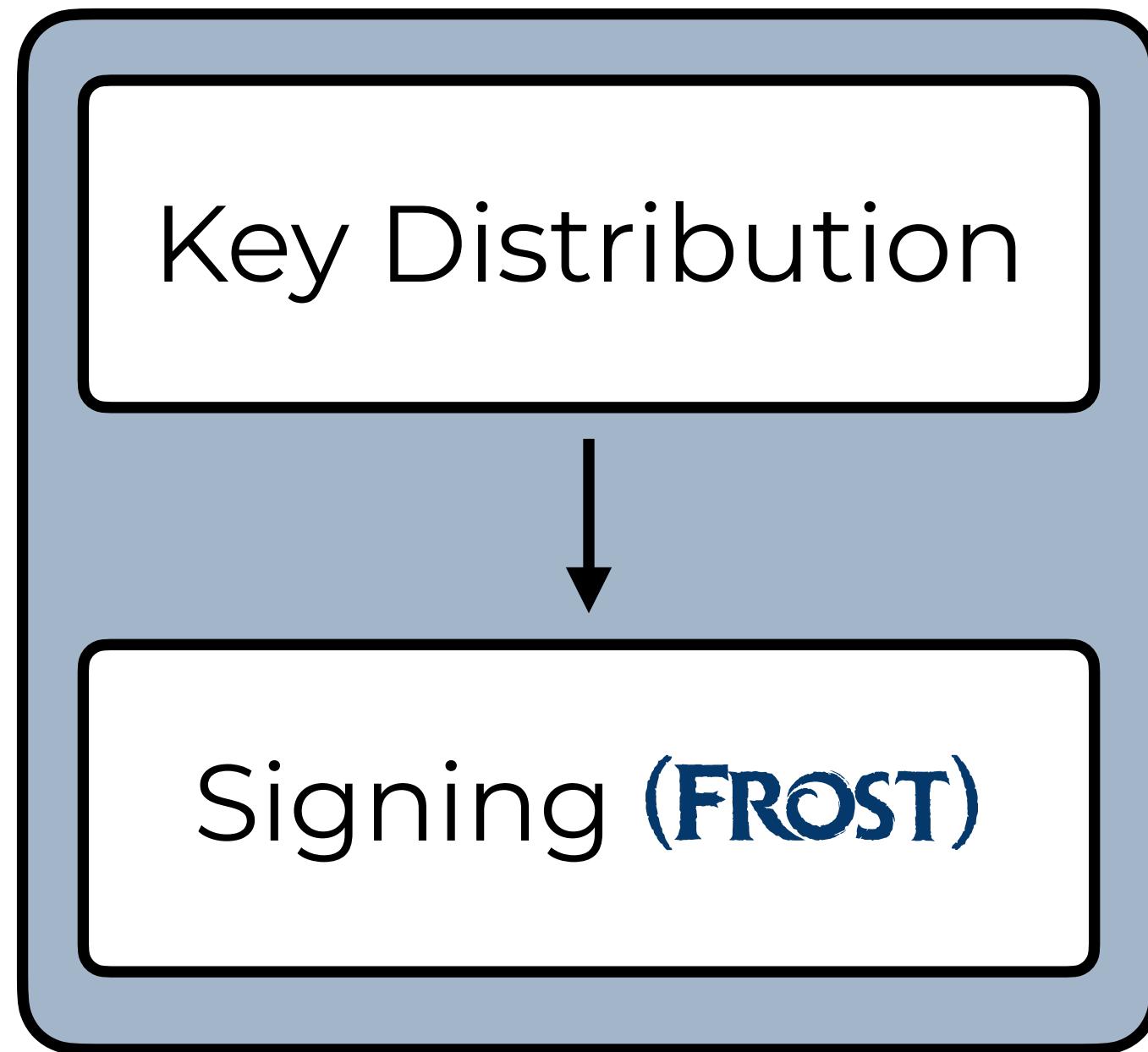
FROST3 SIGNING



The History of Proving FROST Secure



Idealized DKGs [GJKR99,GJKR07,KGS23]



- ✖ No fully simulatable DKG in the dishonest majority setting ($n/2 \leq t - 1$)
- ✖ Simulatable DKGs are not as efficient as the Pedersen DKG

Pedersen DKG with PoP [KG20, CKM21]

$$f_i(Z) = a_{i,0} + a_{i,1}Z + \dots + a_{i,t-1}Z^{t-1}$$

$$f(Z) = \sum_{i=1}^n f_i(Z)$$

$$x_i = f(i), pk = g^{f(0)}$$

$$A_{i,k} = g^{a_{i,k}}$$

$$\text{PoP} = (R, s)$$



Additive sharing of a secret key

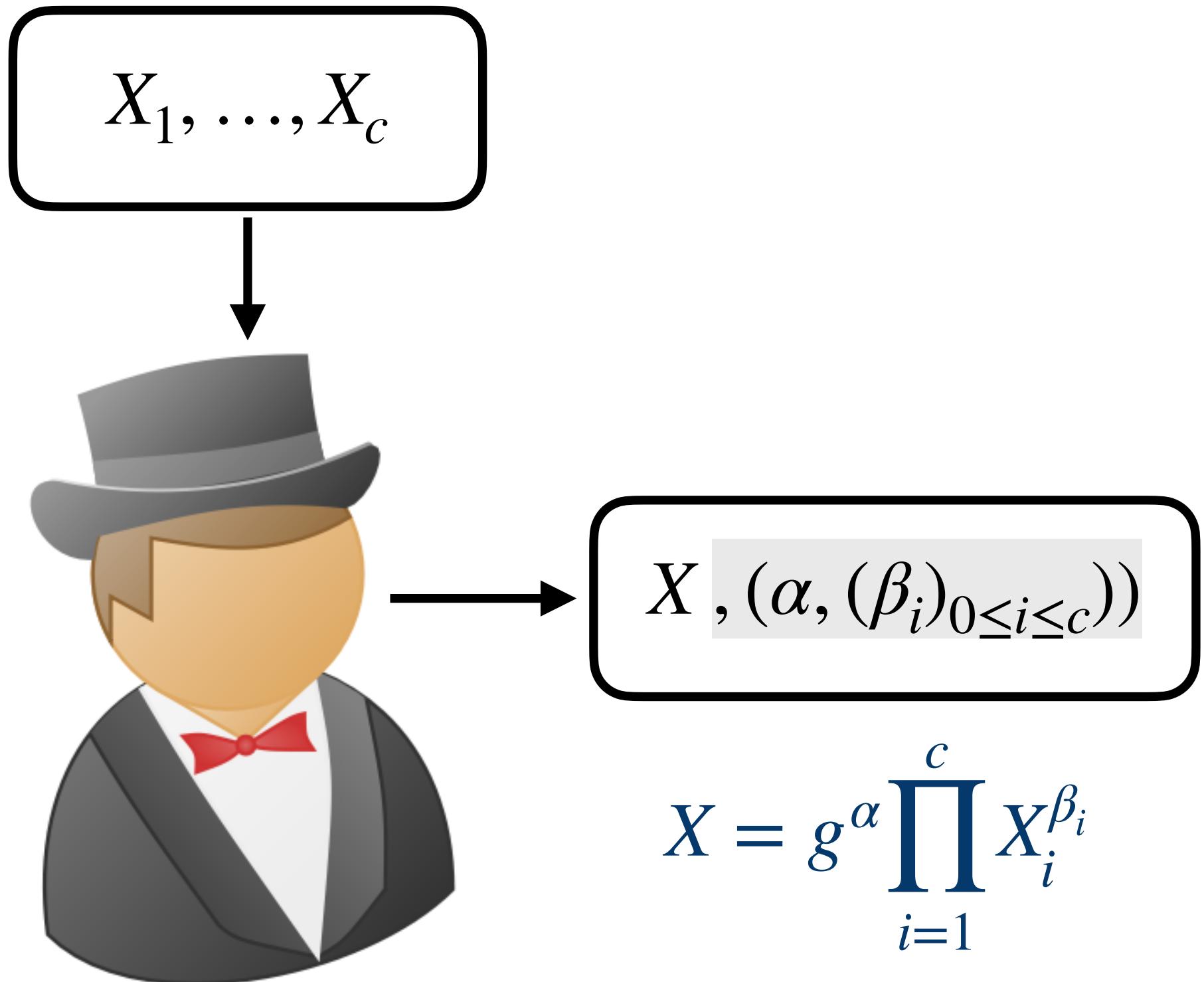
$$x = x_1 + \dots + x_n = a_{1,0} + \dots + a_{n,0}$$



Not fully simulatable DKG due to a possible shift [GJKR06]

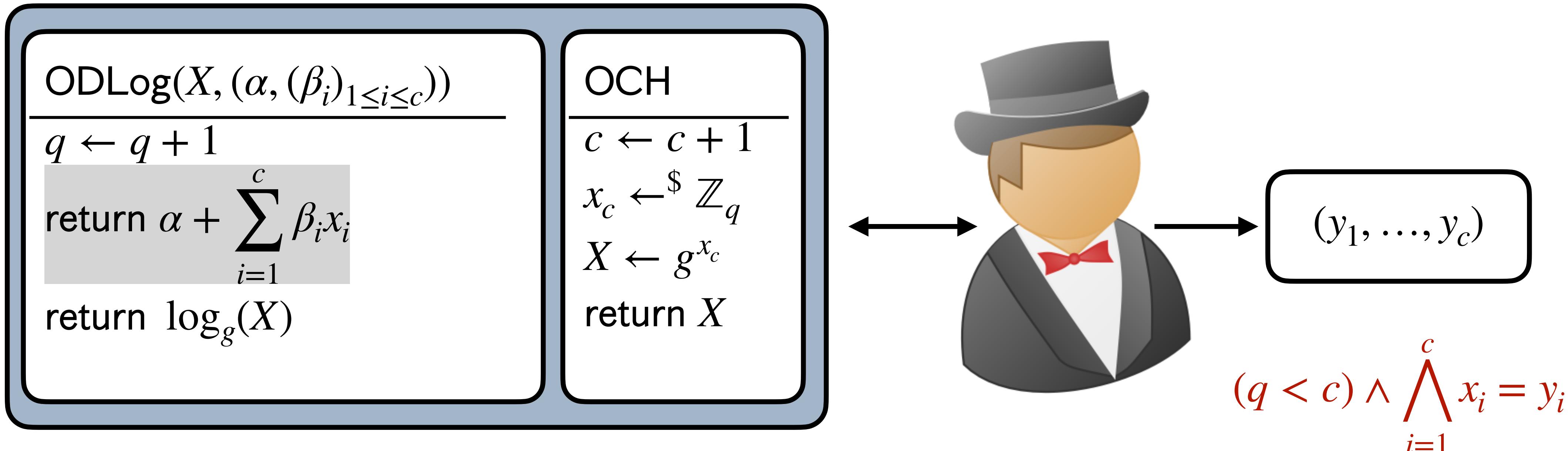
The AGM

[PV05, FKL18]



- ✖ “Hardness in the AGM may not imply hardness in the GGM”
[KZZ22]
- ✖ “[...] The AGM is restricted to the Type Safe model games.” [Zha22]

AOMDL Assumption [NRS21]



From FROST to OLAF

(HOPEFULLY) ONE LAST FROST

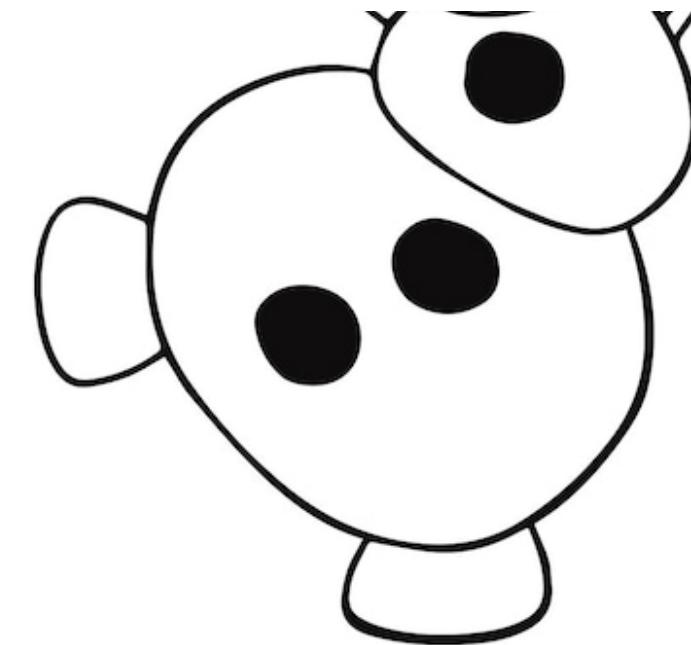


From FROST to OLAF



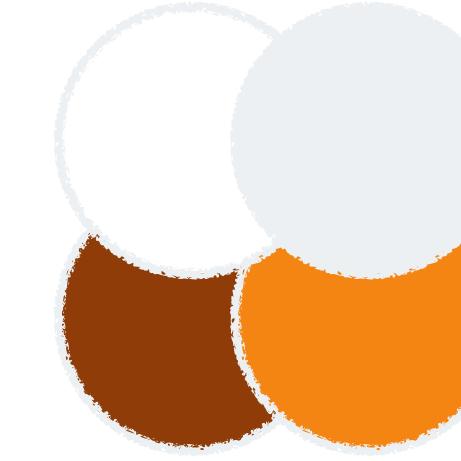
FROST3

+



(Simplified)
Pedersen DKG

+



=



Avoid the AGM
and idealized DKGs

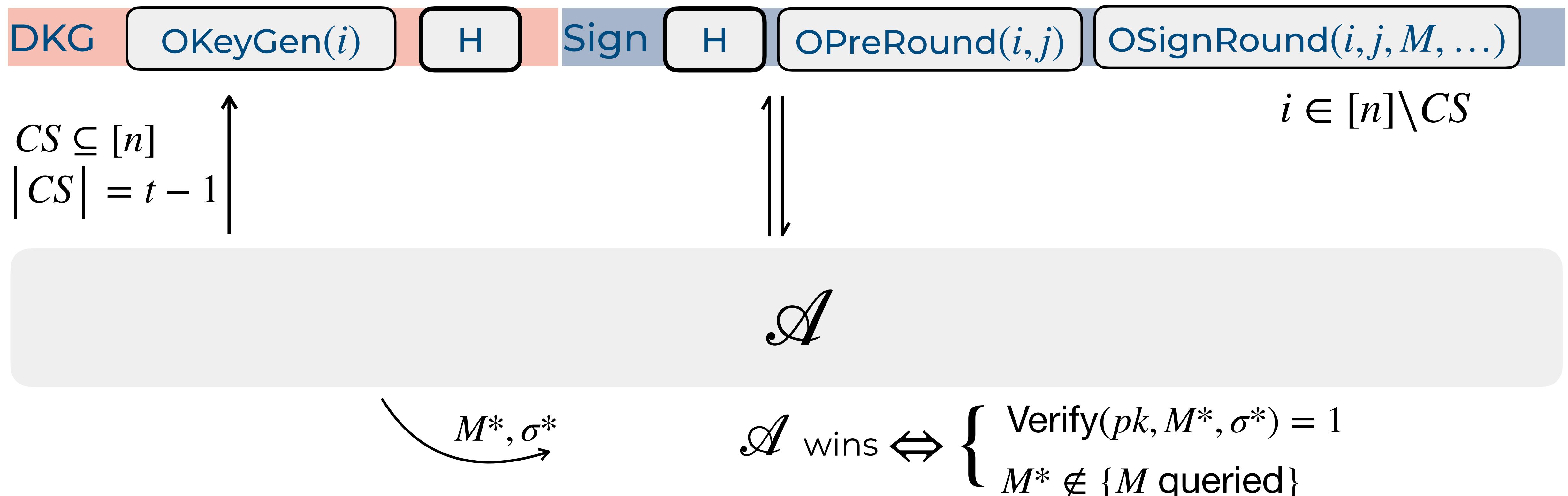
Security Proof for OLAF

V-543-25

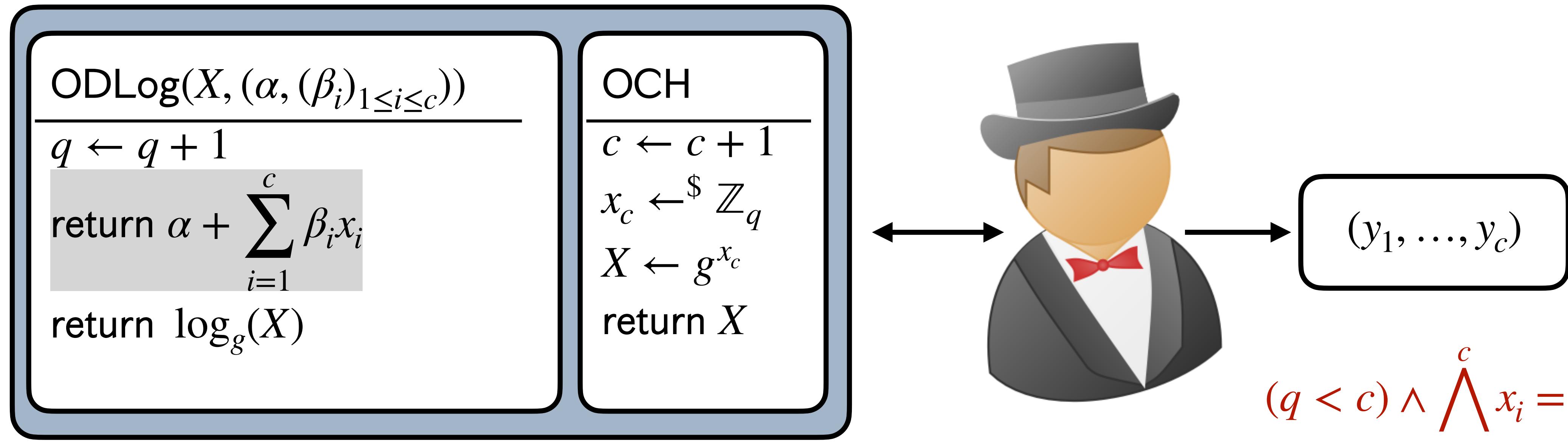
Our Unforgeability Model

Similar to **TS-UF-0** of [BTZ22]

An honest party accepts NO signing queries **before** key setup (DKG) has finished for it



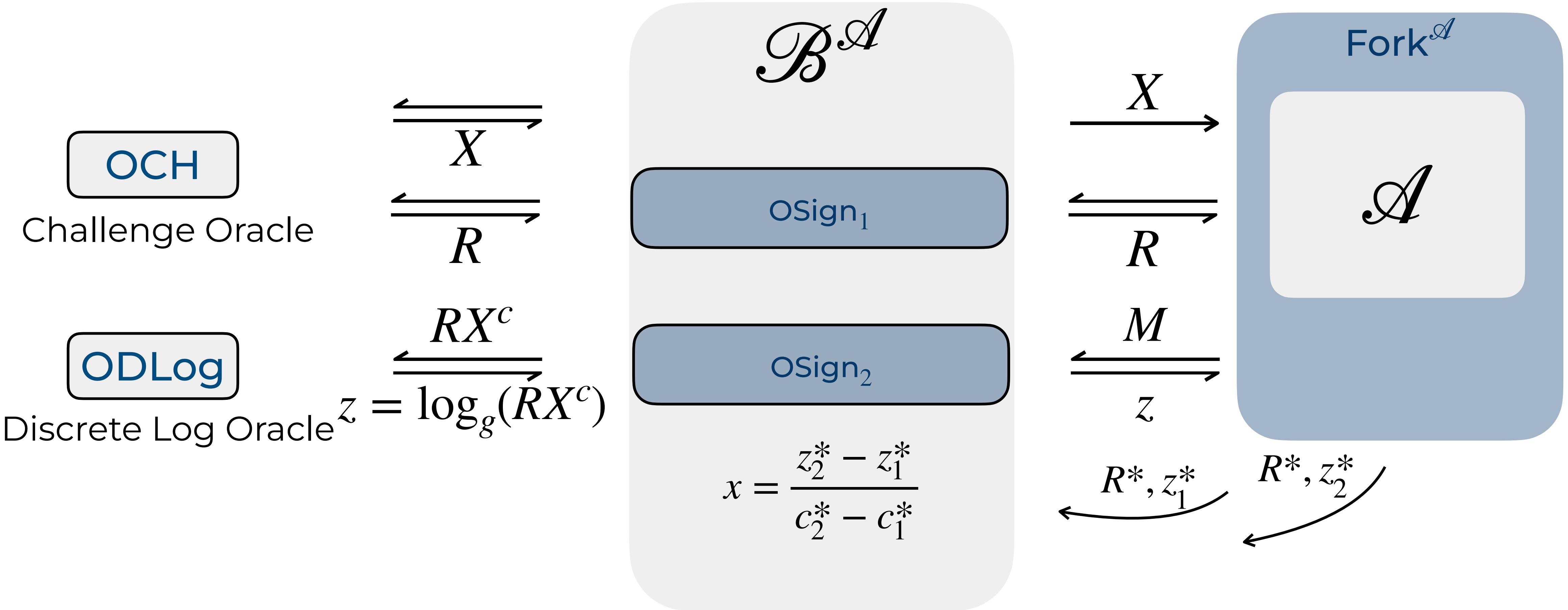
AOMDL Assumption [NRS21]



Proper (A)OMDL proof has less DLog queries than challenges.

Counting DLog queries could be the most tricky part.

AOMDL Proof for Single-Signer Schnorr



$$\#\text{ODLog} = \#\text{OSign}_2 \leq \text{OSign}_1 = \#\text{OCH} - 1$$

AOMDL Proof for Threshold Schnorr

Proof of single-signer Schnorr is well understood

- 1 Simple simulation of signing queries
- 2 Rely on forking [PS00], [BN06], optimal tightness [Seurin11, FJS19]

AOMDL Proof for Threshold Schnorr

Techniques do NOT carry over to the threshold setting

1

Using **two** nonces D, E instead of one nonce (see FROST [KG20])

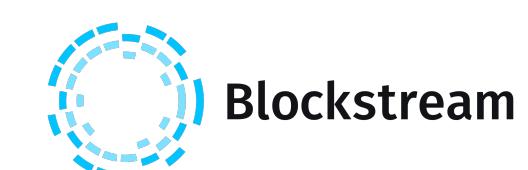
To answer two signing queries on **two executions** for same D, E

2

Consider **Pedersen DKG**

Reduction needs to know each x_i in $\sum_i x_i = x$ instead of just x to solve AOMDL

x_i 's: signing key shares



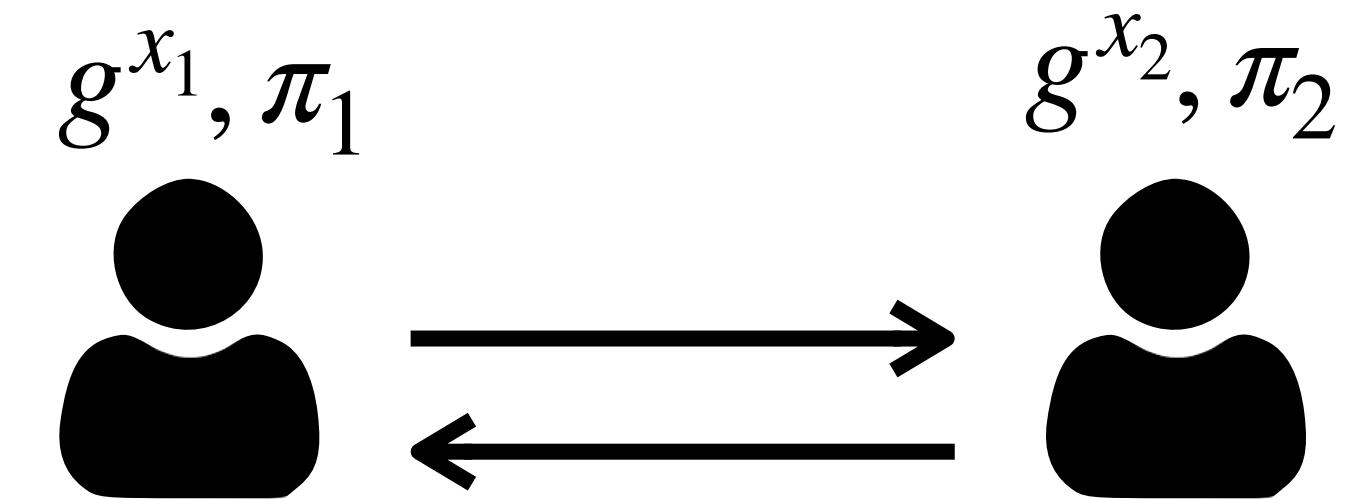
We only have enough DLog queries for **two** executions of signing.

Our Proof Strategy for OLAF

- 1 Use Pedersen DKG with **proofs of possession (PoPs)** as in [CKM21]

PoP is Schnorr-like proof of knowledge

- 2 Use a novel **forking proof technique** to avoid AGM



x_i 's: signing key shares

π_i 's: proof of possession

g^{x_i} 's: public key shares

Existing Forking Techniques: Overview

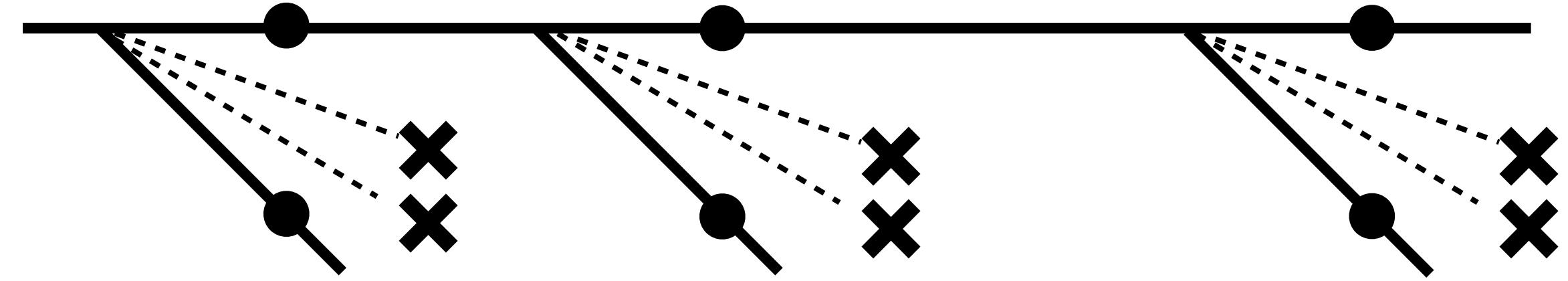


[Bellare-Neven-05] bi-forking lemma $\epsilon' = \epsilon^2/q$

Existing Forking Techniques: Overview

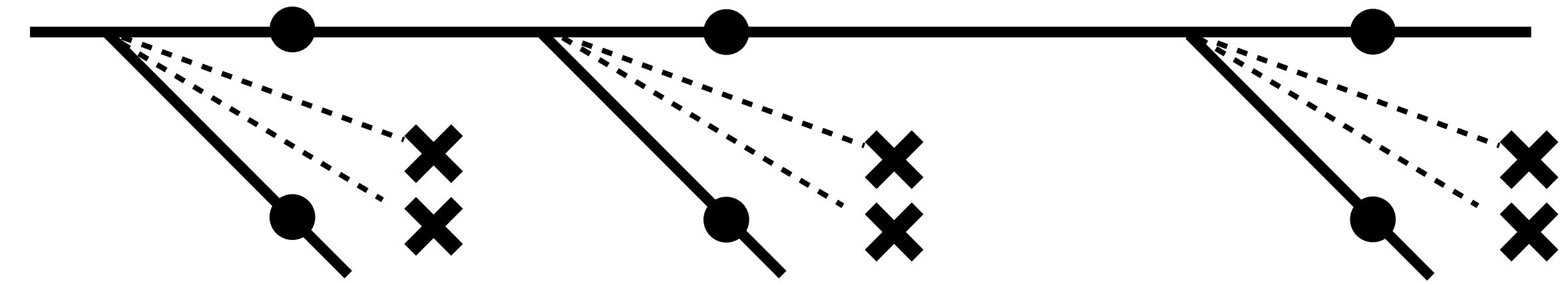


[BN06] bi-forking lemma $\epsilon' = \epsilon^2/q$



[BCJ08] multi-forking lemma $\epsilon' = \epsilon/8$

Existing Forking Techniques: Overview



[BN06] bi-forking lemma $\epsilon' = \epsilon^2/q$

Only two executions

Multiple extraction (in composition) loses 2^t success probability

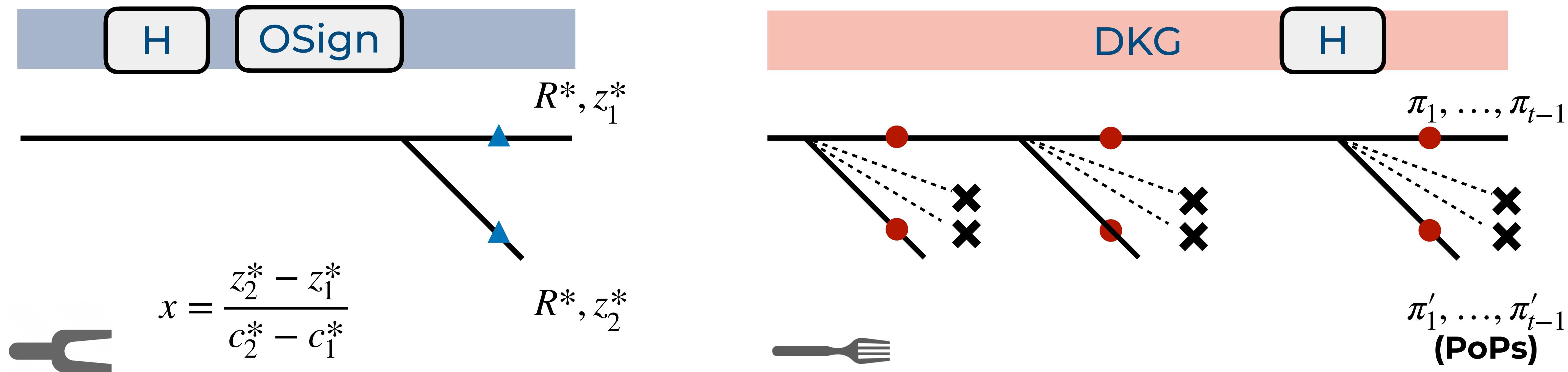


[BCJ08] multi-forking lemma $\epsilon' = \epsilon/8$

Multiple extraction points

Polynomial number of executions

Existing Forking Techniques: In Our Proof

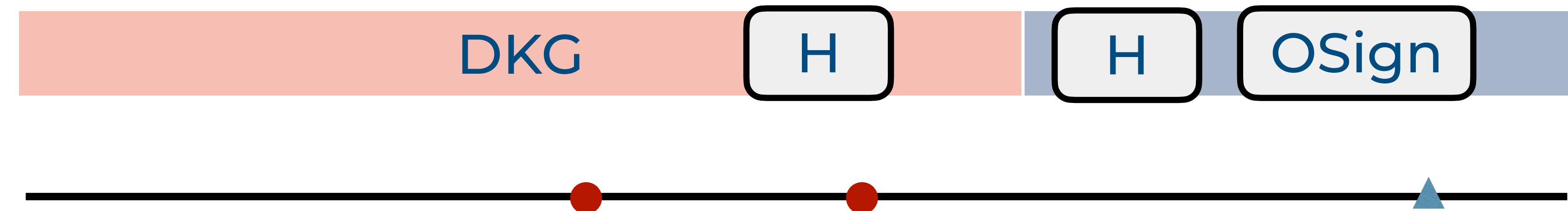


[BN06] bi-forking lemma $\epsilon' = \epsilon^2/q$ [BCJ08] multi-forking lemma $\epsilon' = \epsilon/8$

→ Can extract x from forgeries

→ Can extract x_1, \dots, x_{t-1} from PoPs

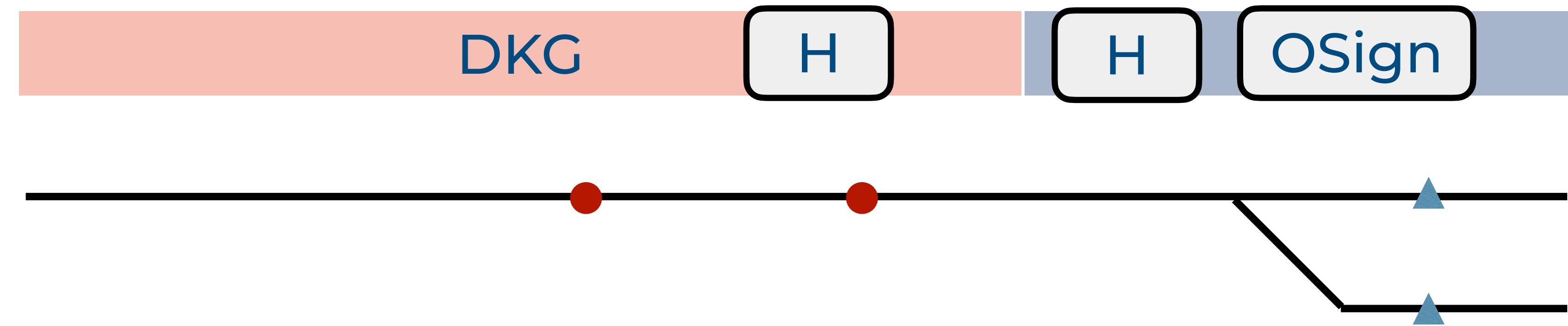
The Mixed Forking



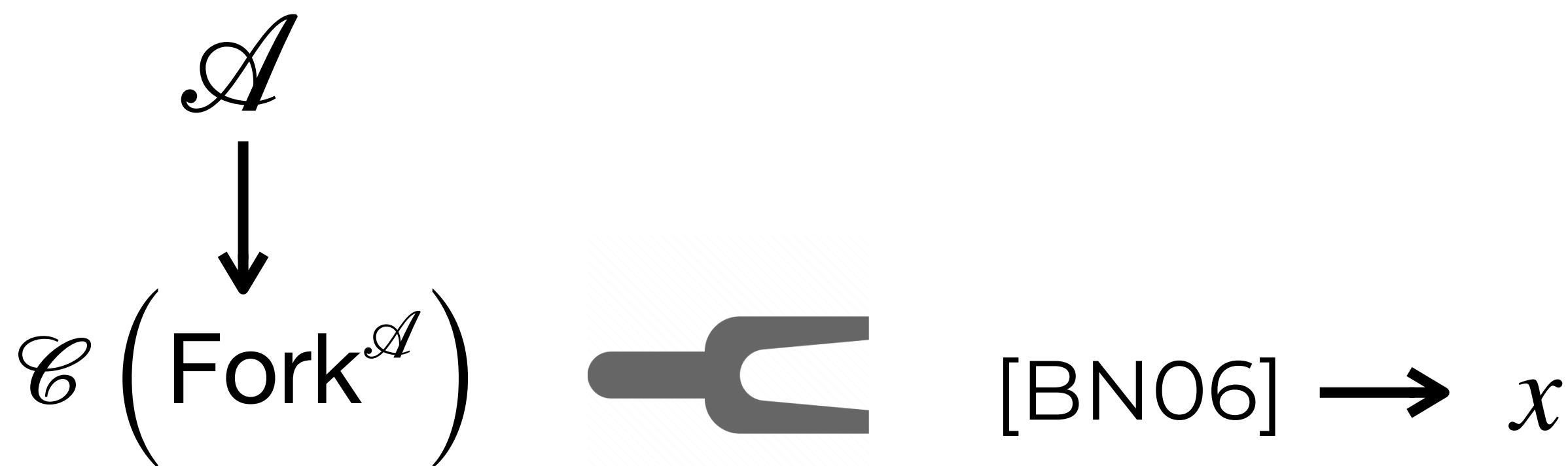
Two-step approach



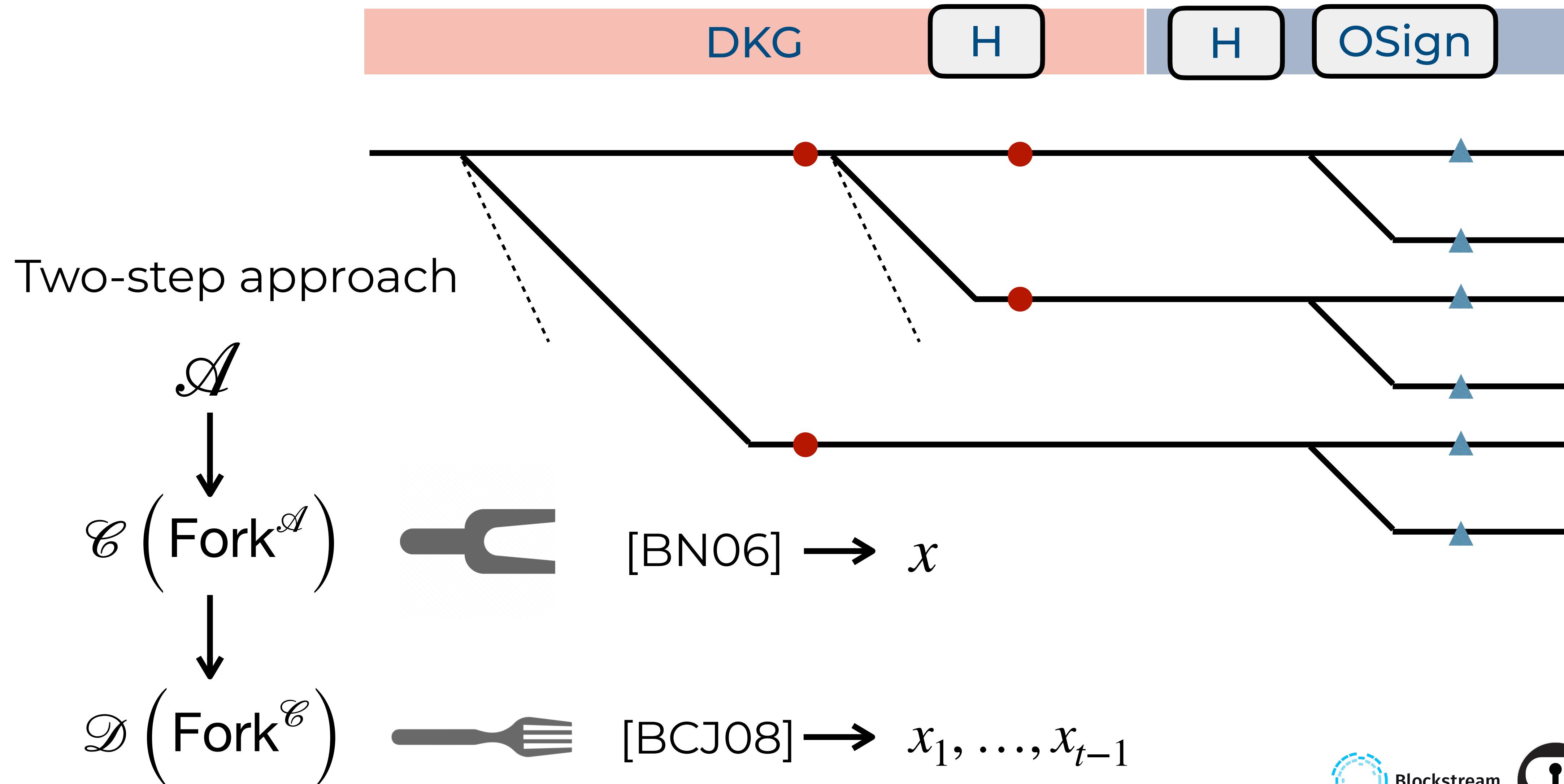
The Mixed Forking



Two-step approach

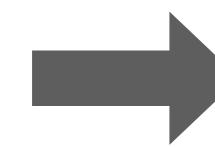
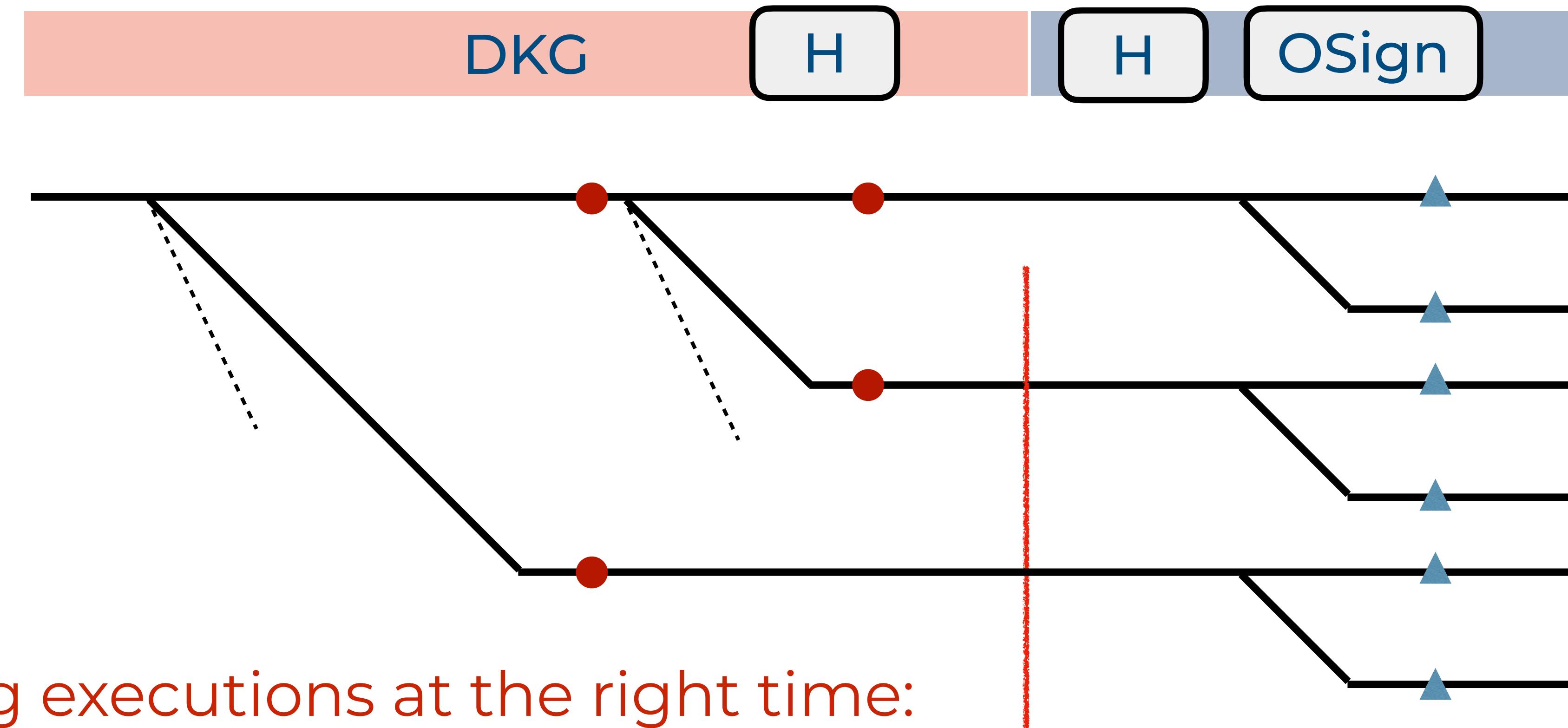


The Mixed Forking



We only have enough DLog queries for **two** executions of signing.

Reducing Number of DL Queries

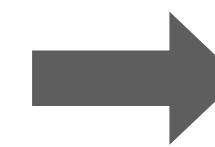
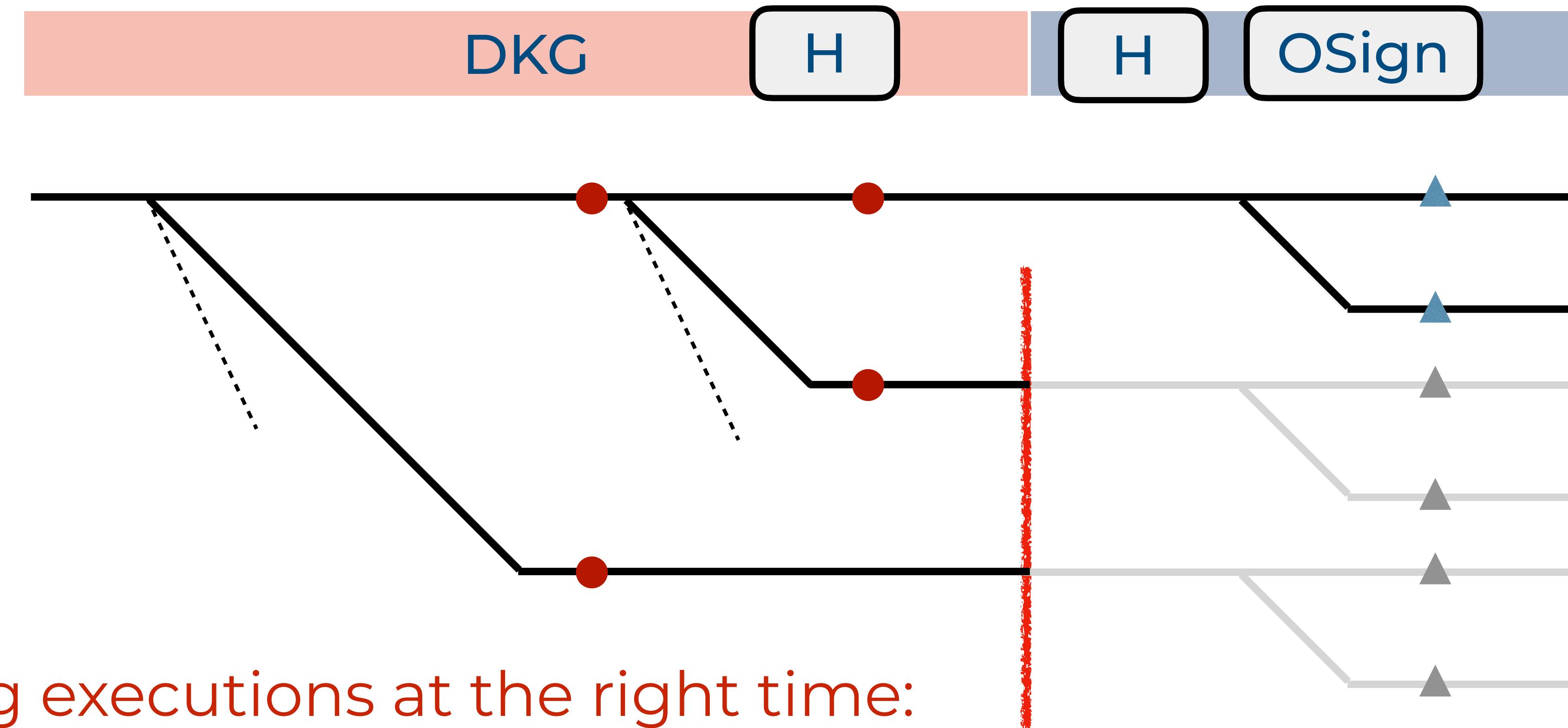


Pruning executions at the right time:

right after the DKG completion

then $\mathcal{D}' \approx \mathcal{D}$ but only **two executions** of \mathcal{A} needs signing simulation

Reducing Number of DL Queries



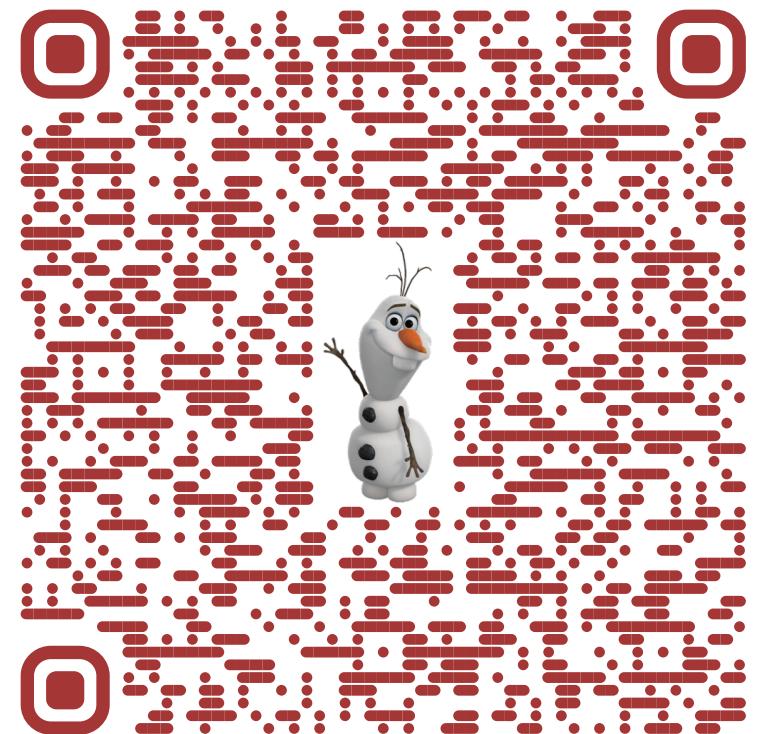
Pruning executions at the right time:

right after the DKG completion

then $\mathcal{D}' \approx \mathcal{D}$ but only **two executions** of \mathcal{A} needs signing simulation

Our Conclusion

- We proved the unforgeability of **FROST3** with probability $\epsilon' \approx \frac{\epsilon^2}{8q}$ in expected $\text{poly}(T)$ time
 - Using a **practical DKG** (Simplified Pedersen DKG + PoPs)
 - In **AOMDL + ROM**



Our Work:

Practical Schnorr Threshold Signatures without the Algebraic Group Model

Hien Chu, Paul Gerhart, Tim Ruffing, Dominique Schröder

CRYPTO 2023



For Robustness, see:

ROAST: Robust Asynchronous Schnorr Threshold Signatures

Tim Ruffing, Viktoria Ronge, Elliott Jin, Jonas Schneider-Bensch, and Dominique Schröder

CCS 2023