Introduction to (profiled) side-channel analysis

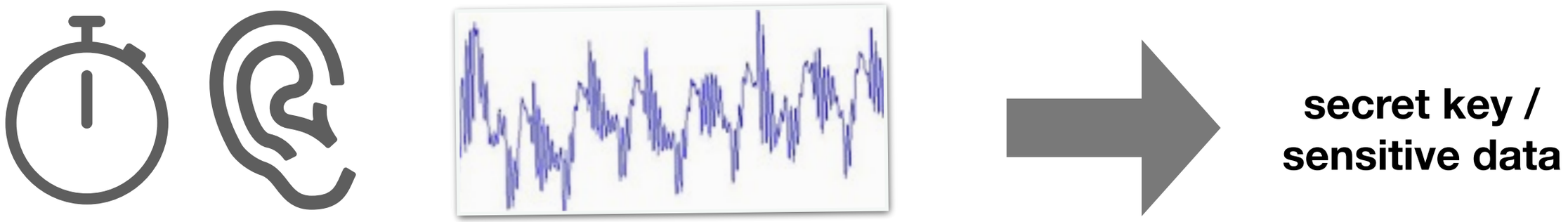
Annelie Heuser

In this talk…

- … back to the basics!!
- details on power / EM leakage (low/high noise scenario)
- how/where to attack AES?
- different attacker models
- overview of side-channel distinguishers
- details on template attack and stochastic approach
- side-channel evaluation metrics
- ….

Side-channel analysis

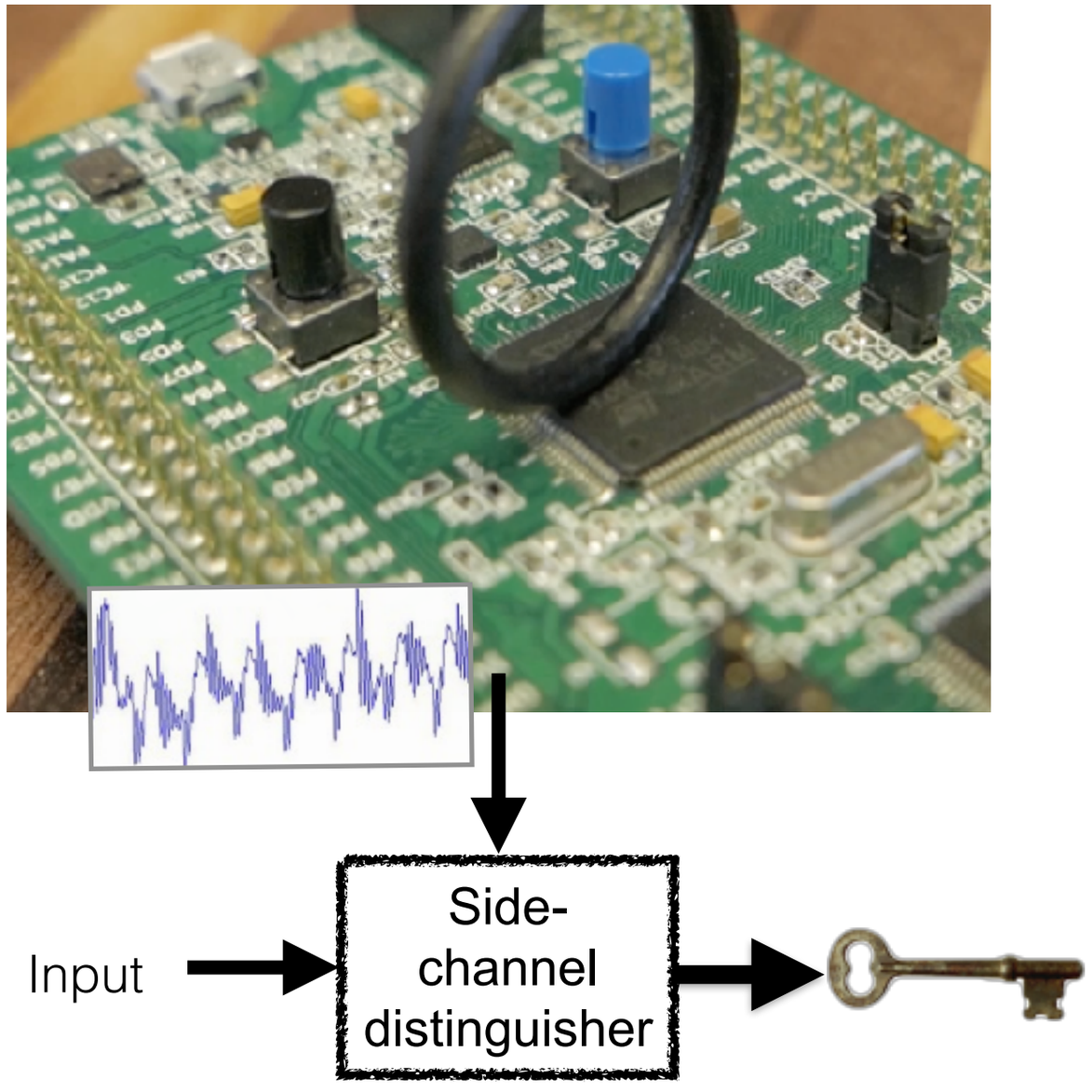
Side-channel information



Side-channel analysis

Invasive hardware attacks, proceeding in two steps:

1) During cryptographic operations capture additional *side-channel* information
   - power consumption/ electromagnetic emanation
   - timing
   - noise, …
2) Side-channel distinguisher to reveal the secret



Side-channel attacks

- …are real in practice

- Beginning 2016: FBI asks Apple to bypass their encryption
- Handful methods to break into the encrypted iPhone
  - software bugs
  - side-channel attacks
  - glitch attack
  - invasive attacks

[edit] (S//NF) Secure key extraction by physical de-processing of Apple's A4 pr

(U) Presenters: [redacted]
AES cryptographic ke
"iDevices". This GID k
stored in system non-v
execution through an
with each new release

(S//NF) The Intelligen
which are public, whi
extract the GID key. If
vulnerabilities, and de
includes the iPhone®

(S) Apple relies on co
manufacturers to supp
design and manufactu
engineering reports h
products. Their data i

[edit] (S//NF) Differential Power Analysis on the Apple A4 Processor

(U) Presenters: [redacted], and [redacted] (U) The Apple A4 processor contains an on-board, AES cryptographic key called the Global ID (GID) that is believed to be shared across all current "iDevices". This GID key is used to un-wrap the keys that decrypt the corresponding boot firmware code stored in system non-volatile memory. Currently, the only way to examine unencrypted boot code is to gain execution through an exploitable software security flaw. However, Apple is quick to address these flaws with each new release of firmware and hardware.

(S//NF) The Intelligence Community (IC) is highly dependent on a very small number of security flaws, many of which are public, which Apple eventually patches. The following presentation will discuss a method to noninvasively extract the GID key from the A4 silicon. If successful, it would enable decryption and analysis of the boot firmware for vulnerabilities, and development of associated exploits across the entire A4-based product-line, which includes the iPhone® 4, the iPod touch® and the iPad®.

(S//NF) Power analysis techniques have proven effective in extracting hardware resident cryptographic information, such as cryptographic keys, from secure processors noninvasively through side-channel methods. We have worked to develop an environment within the iPhone 4 that assists analysts in performing differential power analysis (DPA) attacks against the A4 processor while preserving the functionality of the device. We have studied electromagnetic (EM) emissions that occur during AES operations with the intent of extracting information about the on-chip AES keys. We will discuss the methods used to acquire various measurements from the system and the progress we've made in attempting to extract the GID key from the devices.

Documents released by Snowden: NSA is studying the use of side-channel attacks to break into iPhones

Side-channel attacks

- …are real in practice
- attacking Philips Hue smart lamps
- side-channel attack revealed the global AES-CCM key used to encrypt and verify firmware updates
- insert malicious update: lamps infect each other with a worm that has the potential to control the device

Paper: Eyal Ronen et al, IoT Goes Nuclear: Creating a ZigBee Chain Reaction

Observable leakage

- In this talk: Power/EM as leakage source
  - register writing, loading / storing, computations
  - bytes, bits, nibbles, … (also architecture dependent)
  - coarse grained model: Hamming weight/distance model
  - fine grained model: intermediate states / key values

Side-channel targets

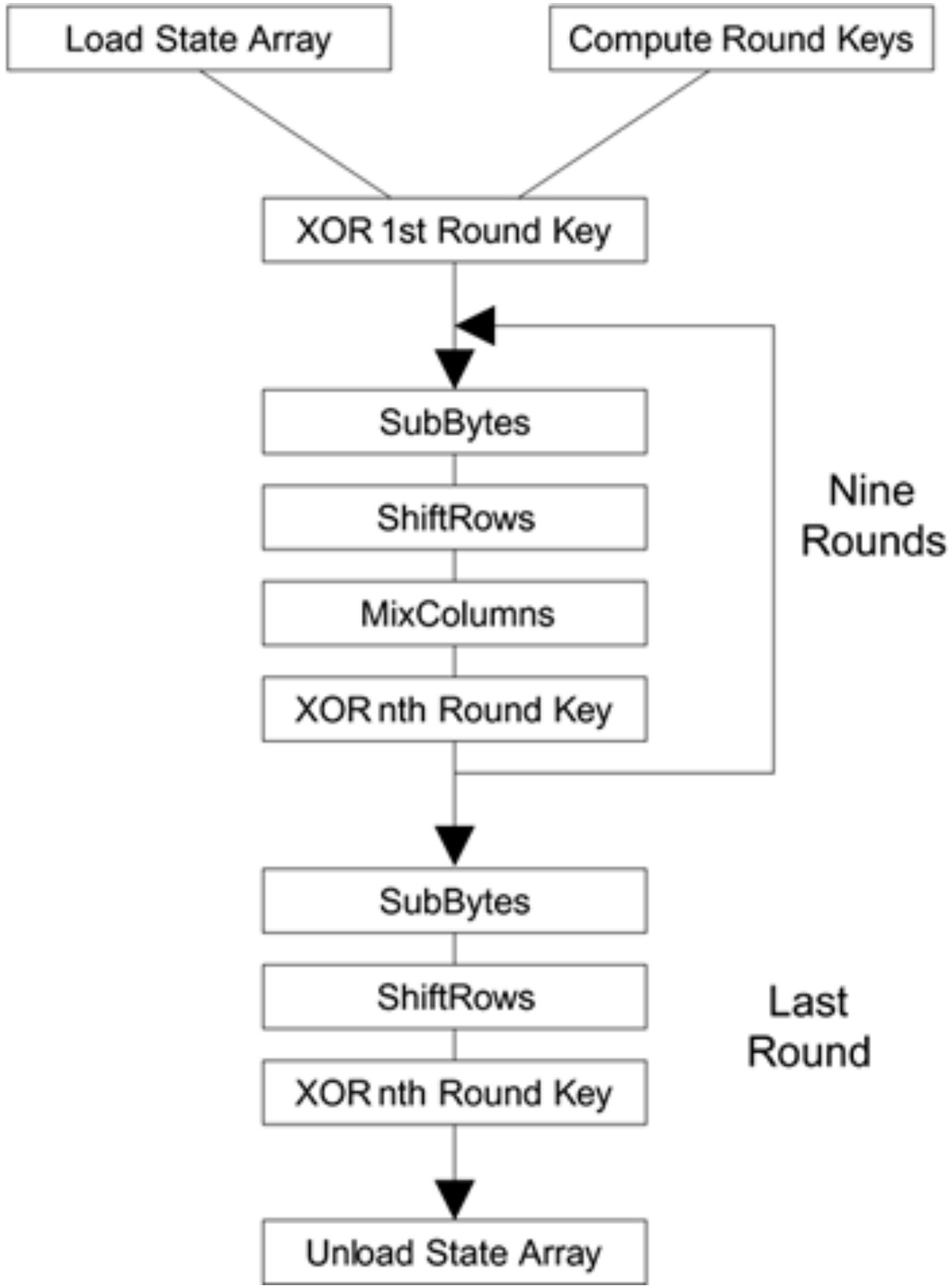
- Symmetric block ciphers
- Asymmetric block ciphers
- Signatures
- Post-quantum schemes
- hash-based message authentication code (HMAC)
- …

Symmetric key crypto

- input: plaintext
- output: ciphertext
- secret key used for encryption and decryption
- block ciphers: AES, lightweight ciphers: PRESENT
- with side-channel information able to reveal secret key

AES

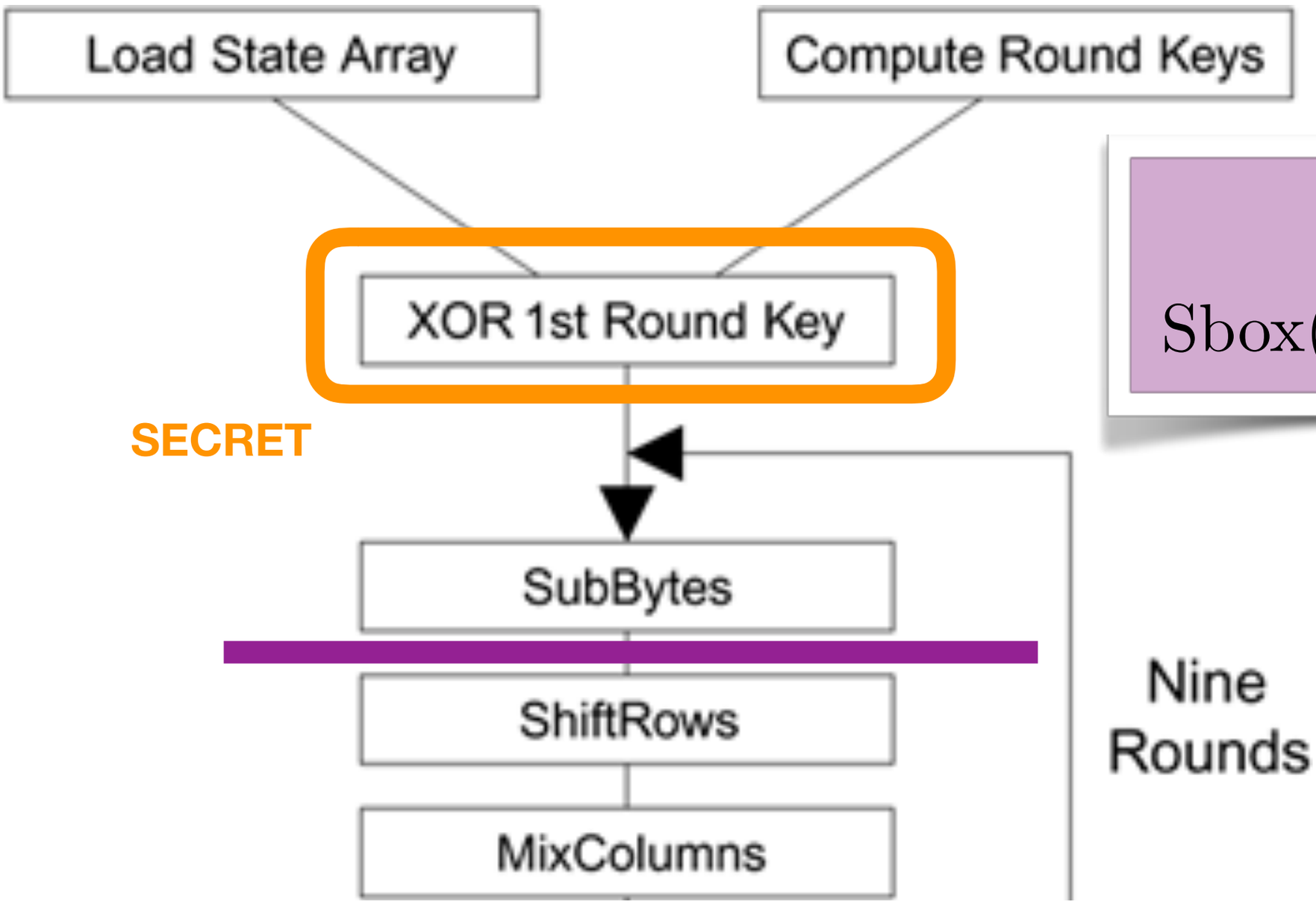
- plaintext/ciphertext: 128-bit
- secret key: 128, 192, 256 bits with 10/12/14 rounds
- each round distinct round key



Side-channel attacks on AES

- secret key: 128, 192, 256 bits (infeasible to iterate on)
- side-channel attacks use divide-and-conquer
- attack each byte independently
- 256 key guesses, iteration easily possible
- on embedded devices typically operating/processing on bytes
- key byte information are mixed using MixColumns operation => attack before!
- Typically first round or last round…

SCA on AES (first round)



label:

$$\text{Sbox}(plaintext \oplus key)$$

SBox and key guesses

- Toy example:
  - 6 plaintext bytes = [24 1 230 50 10 155];
  - 3 key guesses = [1 2 3];

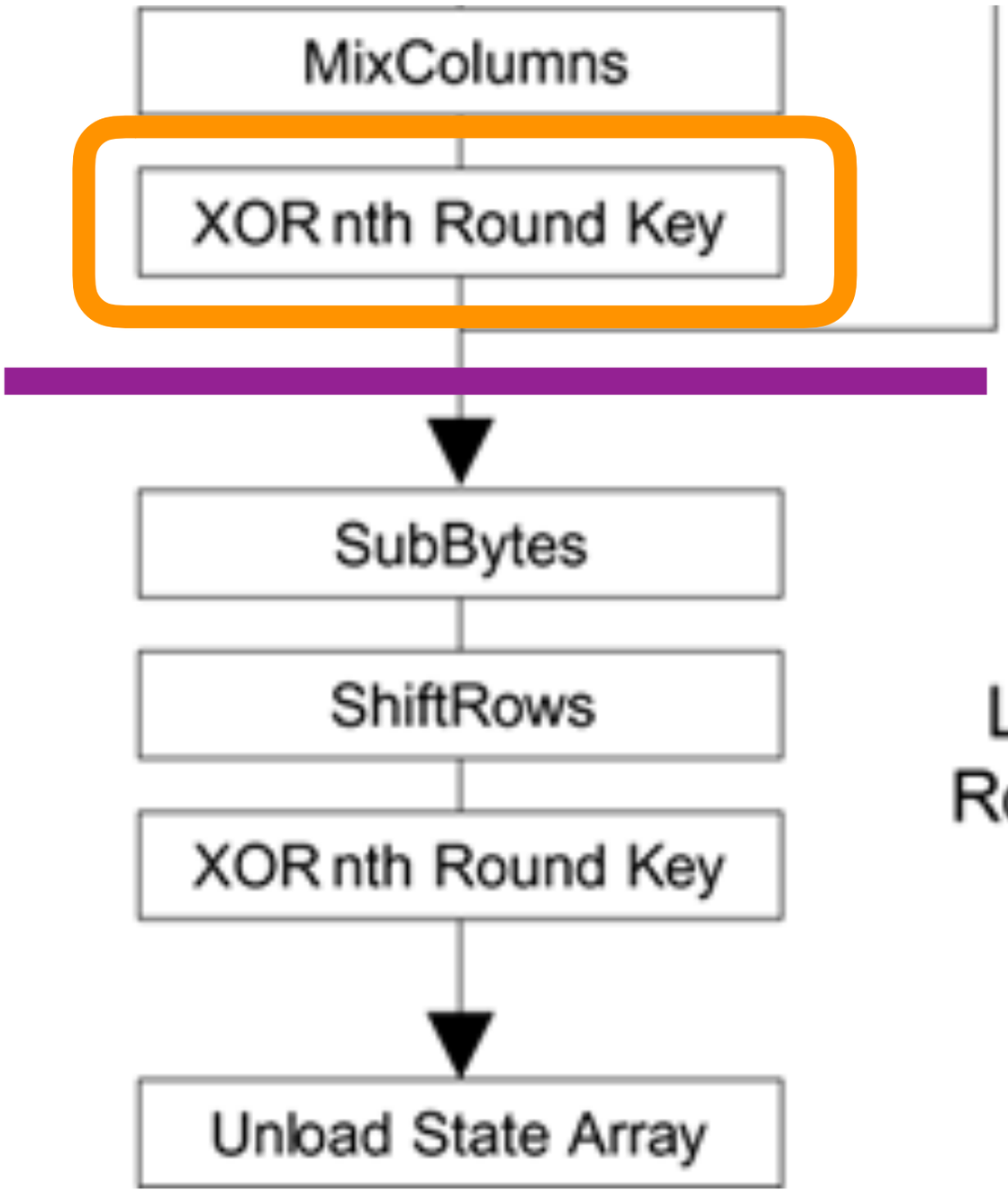
$$plaintext \oplus key$$

$$\begin{bmatrix} 25 & 26 & 27 \\ 0 & 3 & 2 \\ 231 & 228 & 229 \\ 51 & 48 & 49 \\ 11 & 8 & 9 \\ 154 & 153 & 152 \end{bmatrix}$$

$$\text{SBox}(plaintext \oplus key)$$

$$\begin{bmatrix} 212 & 162 & 175 \\ 99 & 123 & 119 \\ 148 & 105 & 217 \\ 195 & 4 & 199 \\ 43 & 48 & 1 \\ 184 & 238 & 70 \end{bmatrix}$$

SCA on AES (last round)



label:

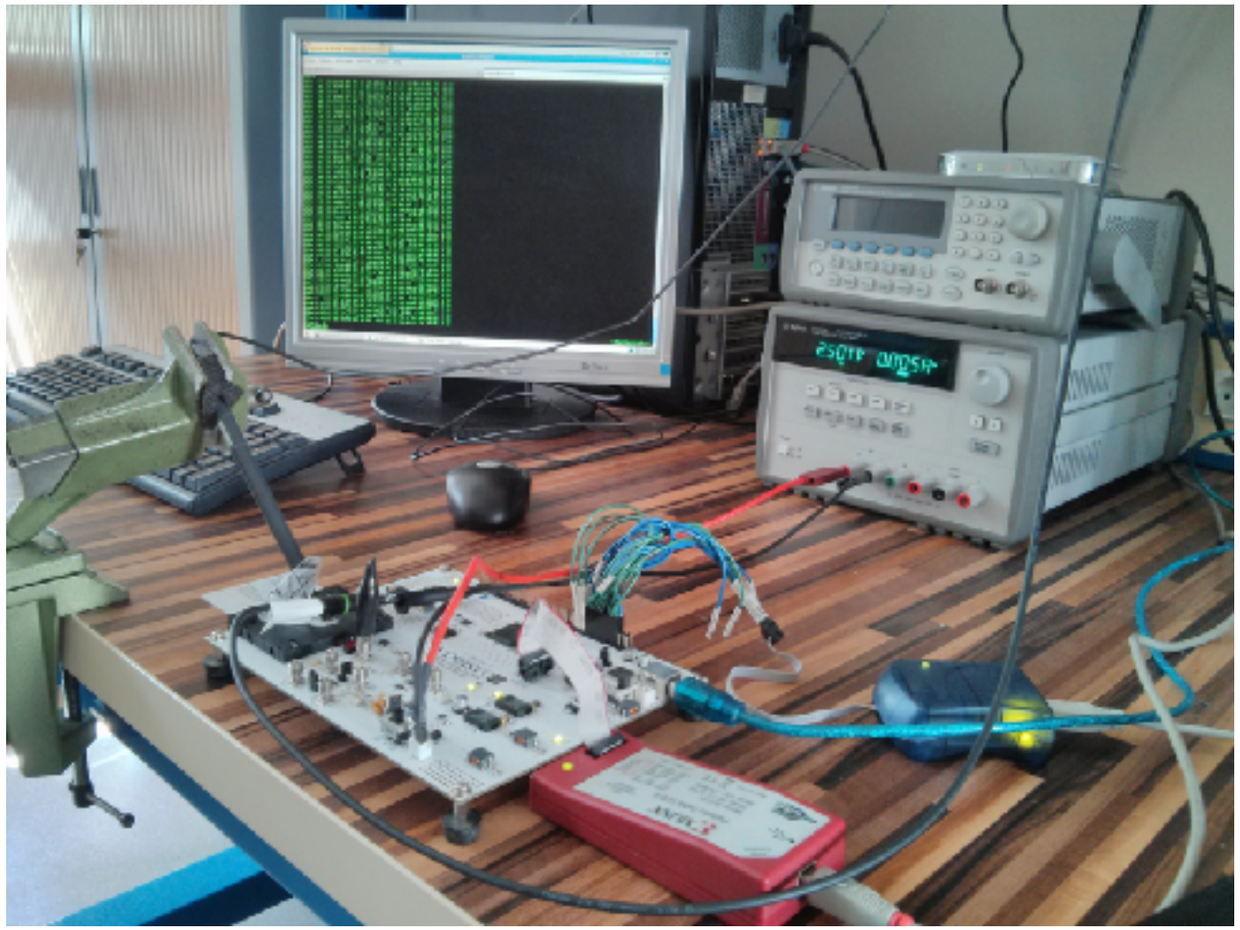
$$\mathrm{Sbox}^{-1}(cipher(byte) \oplus key)$$

or

$$\mathrm{Sbox}^{-1}(cipher(byte) \oplus key) \oplus cipher(byte')$$

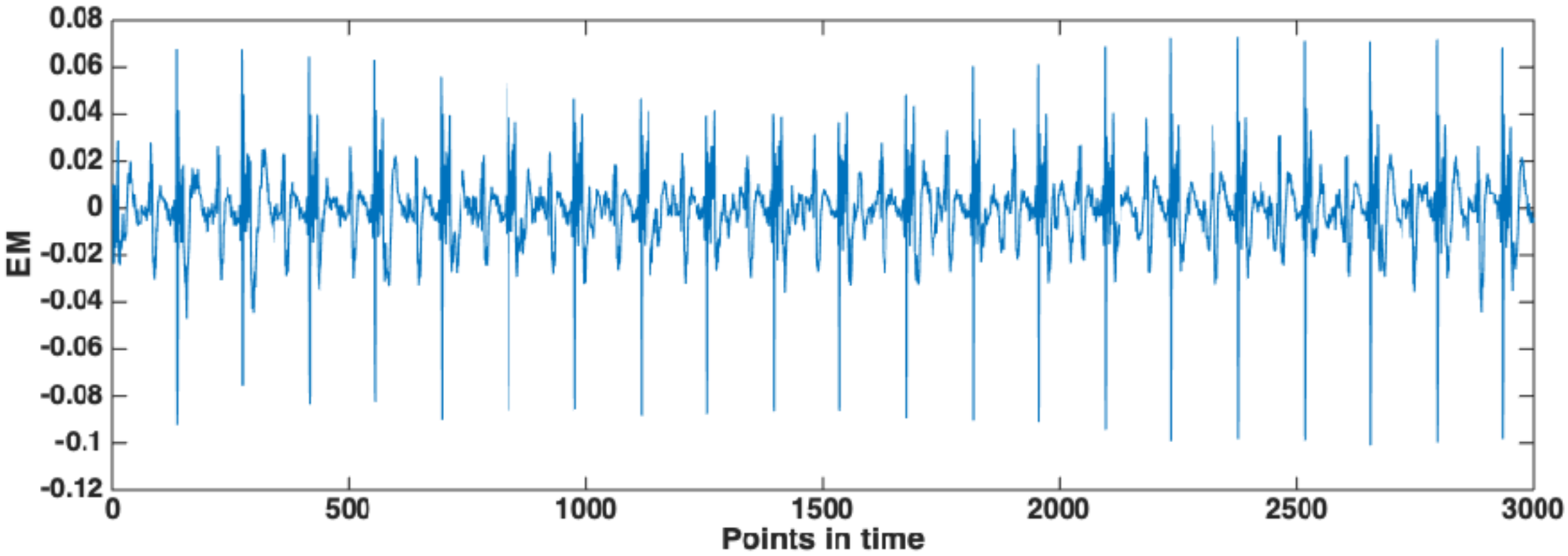
Dataset 1

- Low noise dataset - DPA contest v4 (publicly available)
- Atmel ATMega-163 smart card connected to a SASEBO-W board
- AES-256 RSM
  (Rotating SBox Masking)
- In this talk:
  mask assumed known
- used in this talk:
  1 000 000



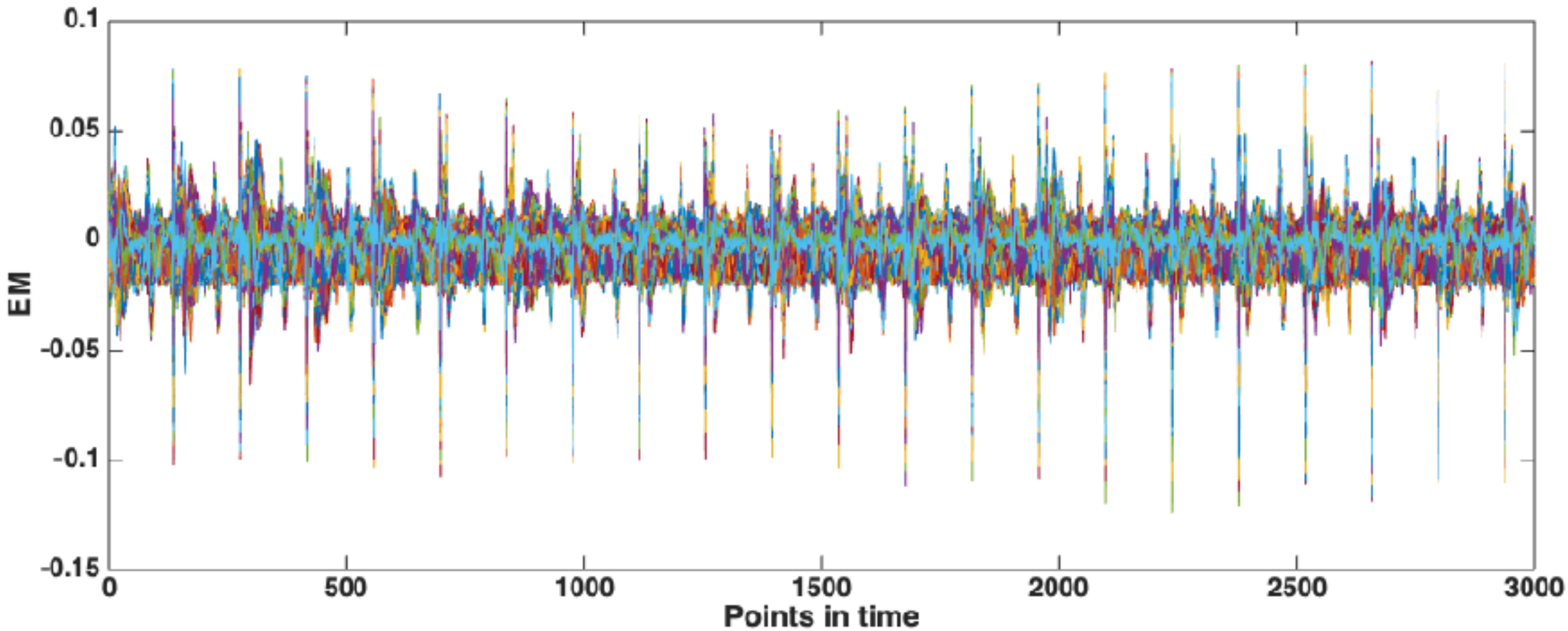
Traces

- Trace length regarding one S-box operation: 3000



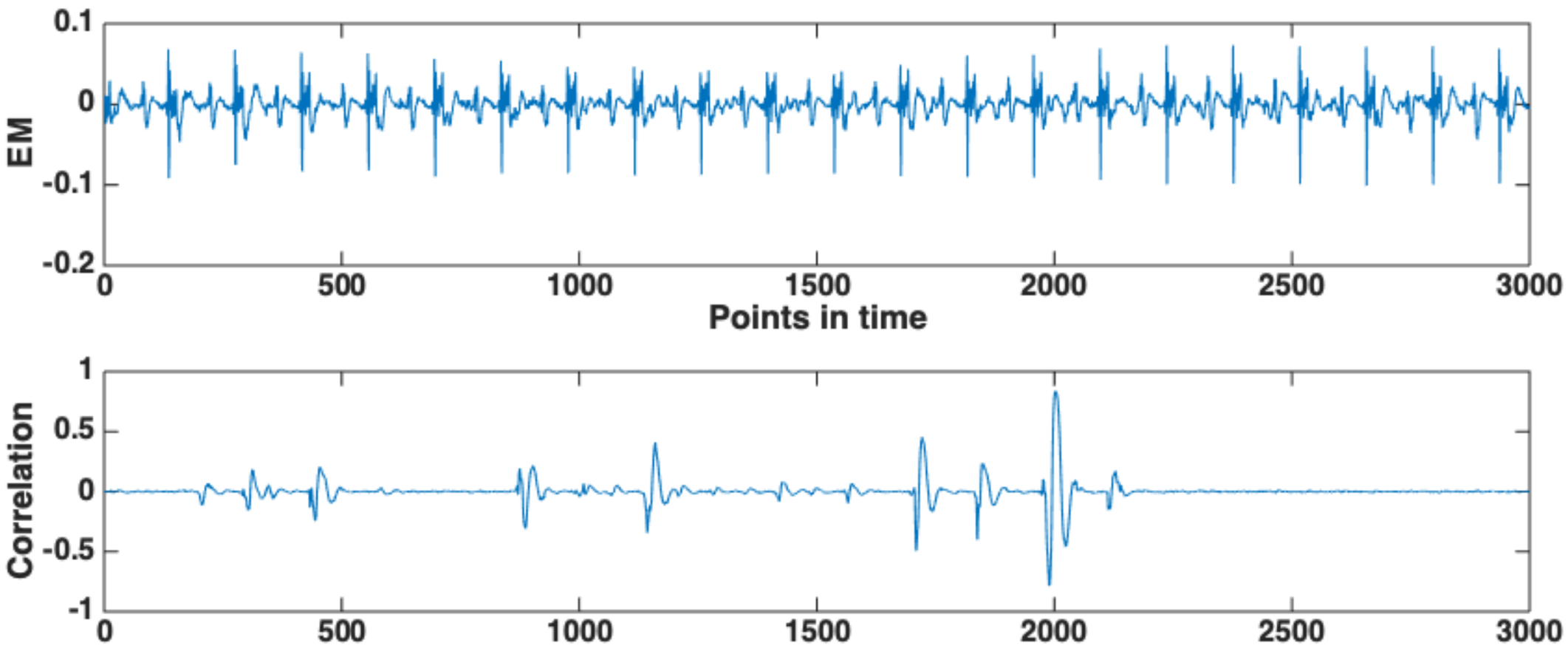
Traces

- Trace length regarding one S-box operation: 3000



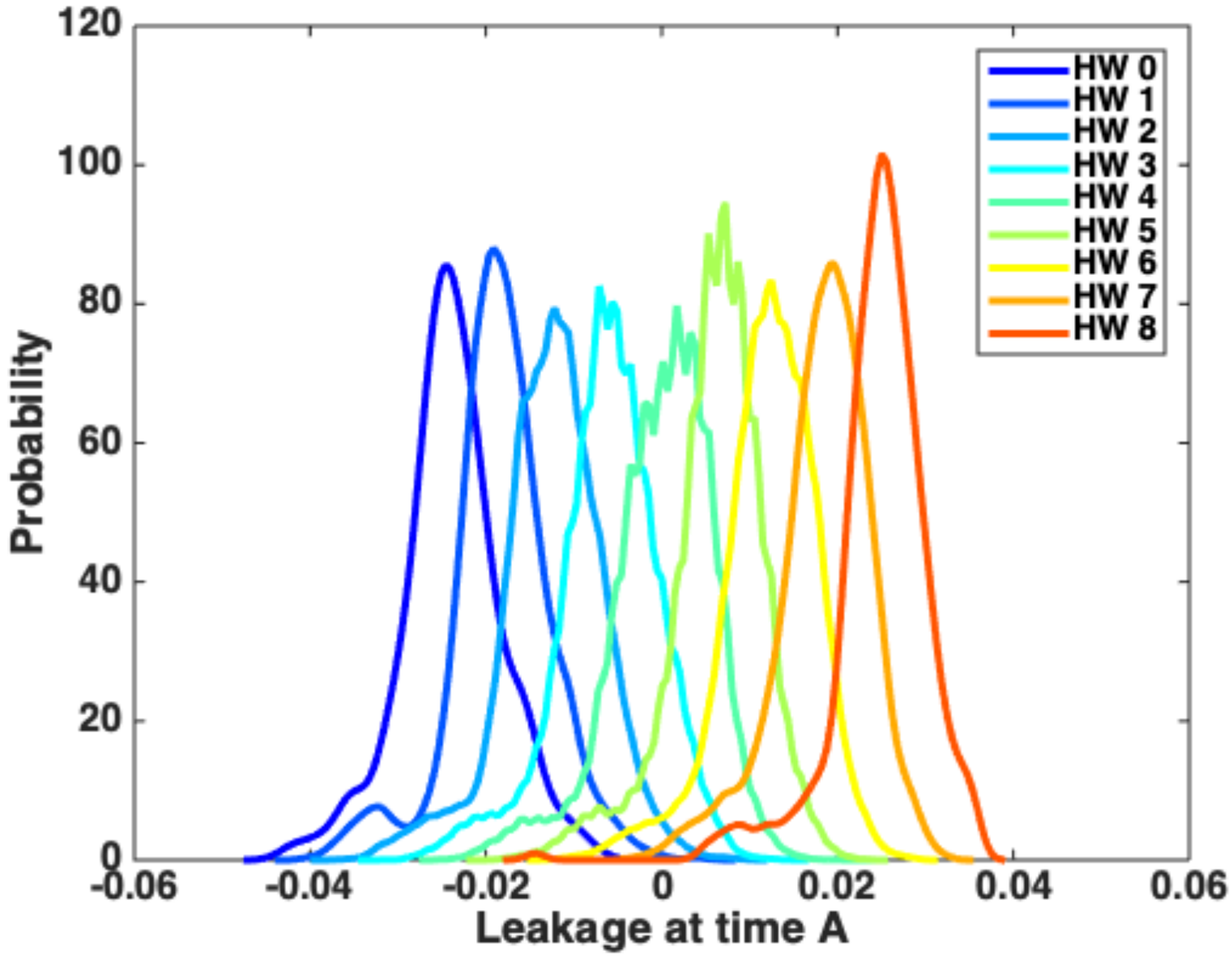
Leakage

- Attack first round
- Correlation between HW of the Sbox output and traces



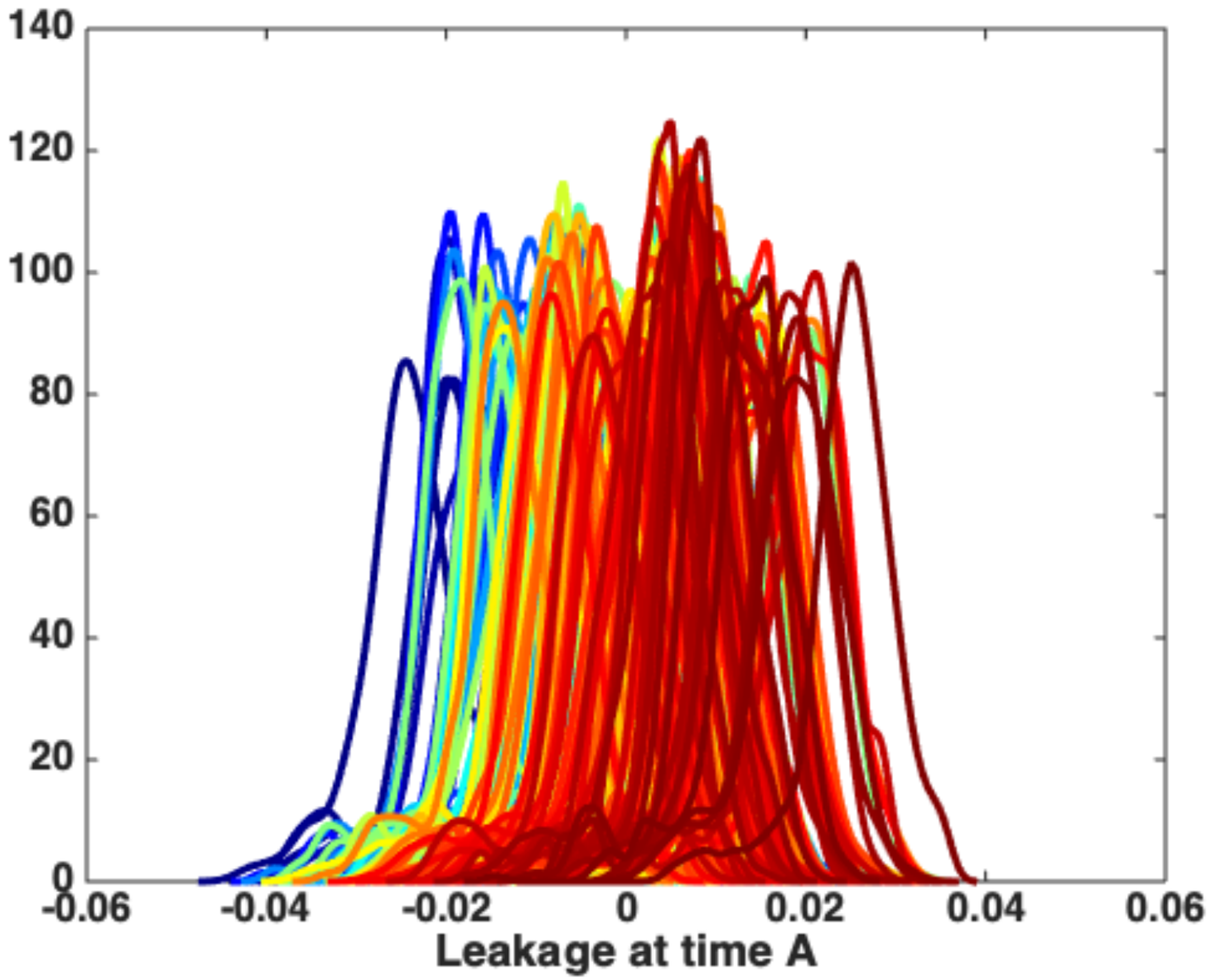
Observable leakage

- HWs of the Sbox output are easily distinguishable
- Indications that the HW model not precise



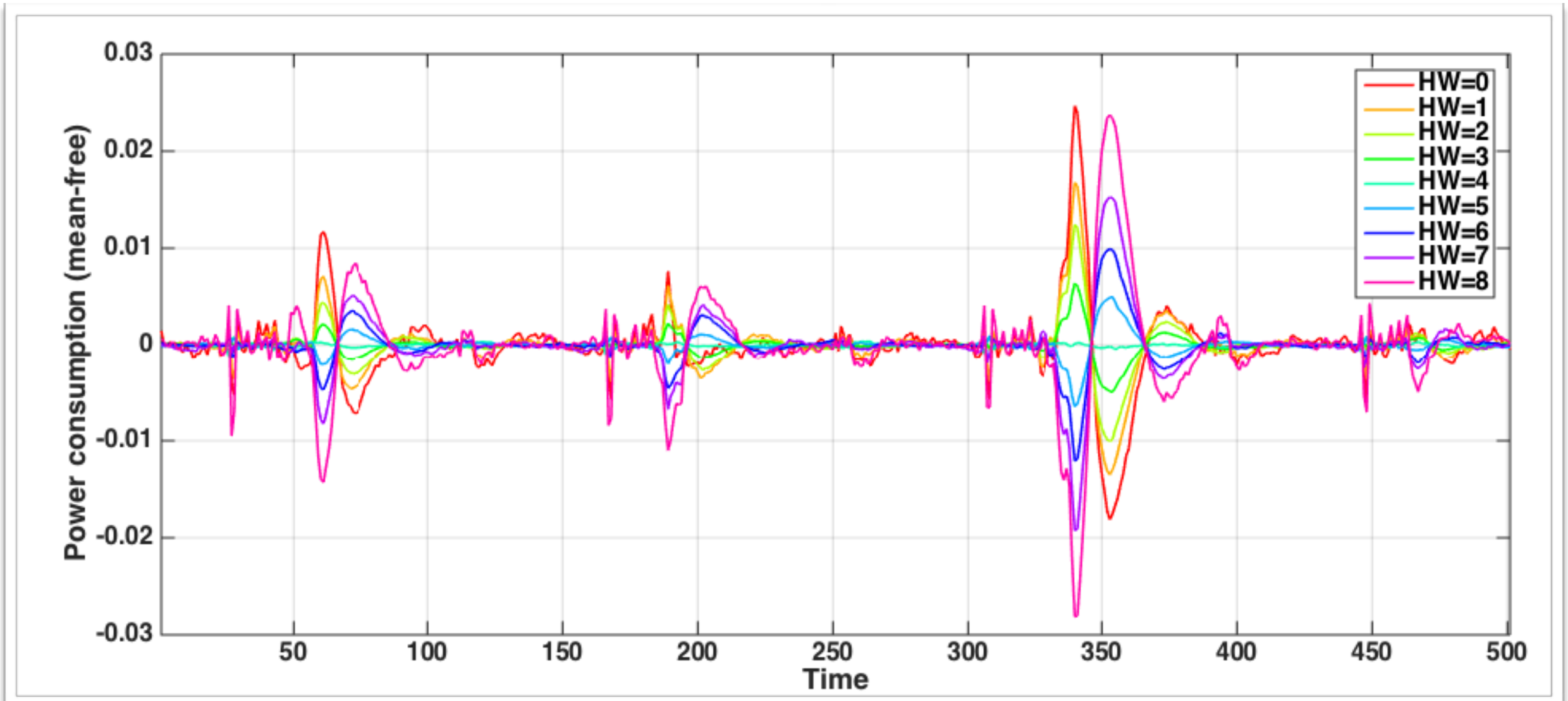
Observable leakage

- Densities according to the Sbox output



Observable leakage

- Hamming weight grouping over time

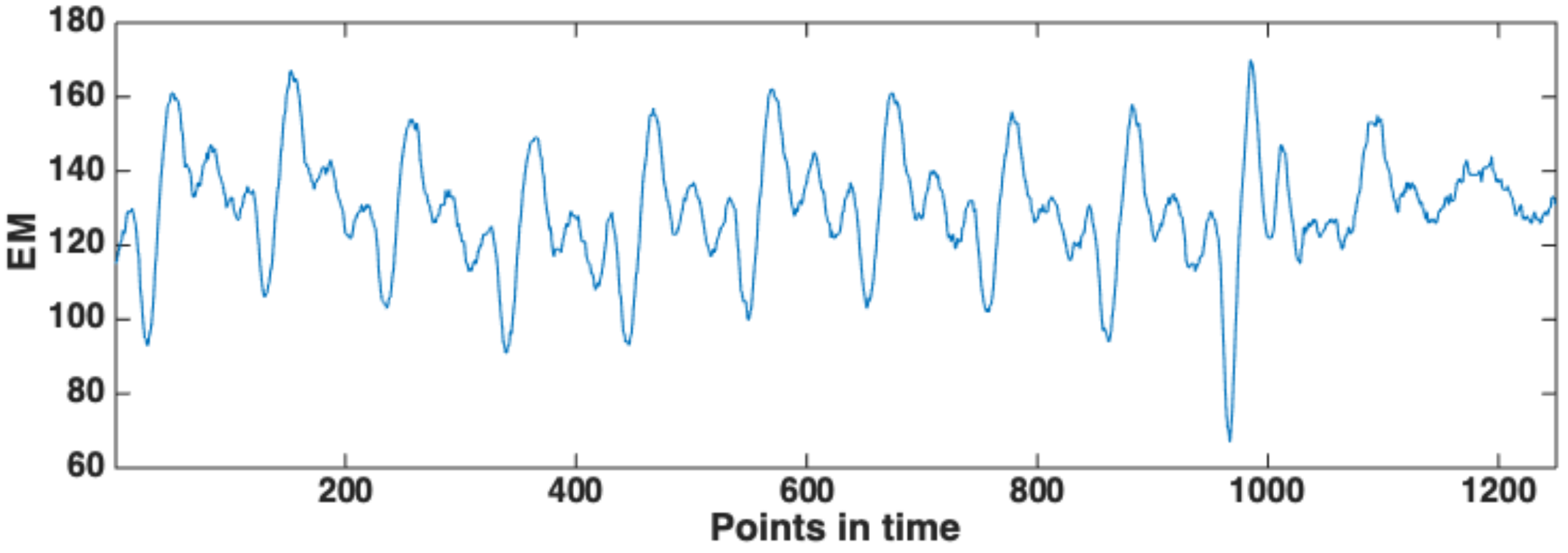


Dataset 2

- High noise dataset (still unprotected!)
- AES-128 core was written in VHDL in a round based architecture (11 clock cycles for each encryption).
- The design was implemented on Xilinx Virtex-5 FPGA of a SASEBO GII evaluation board.
- used in this talk: 1 000 000
- publicly available on github: https://github.com/AESHD/AES HD Dataset

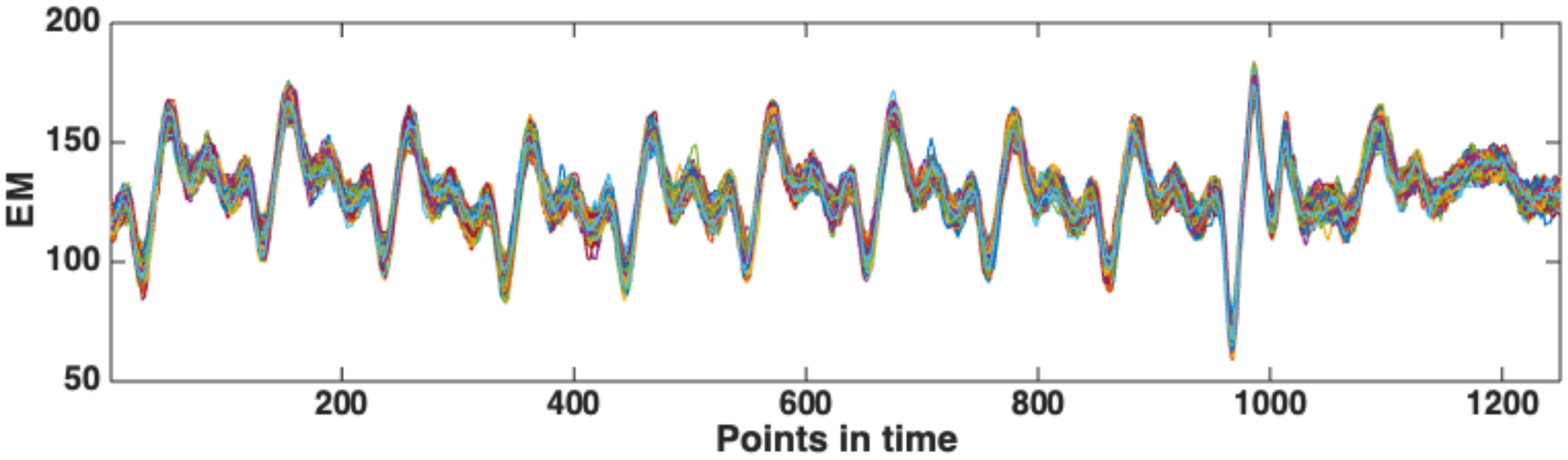
Traces

- Complete trace length: 1250
- Trace length regarding one S-box operation: approx 150



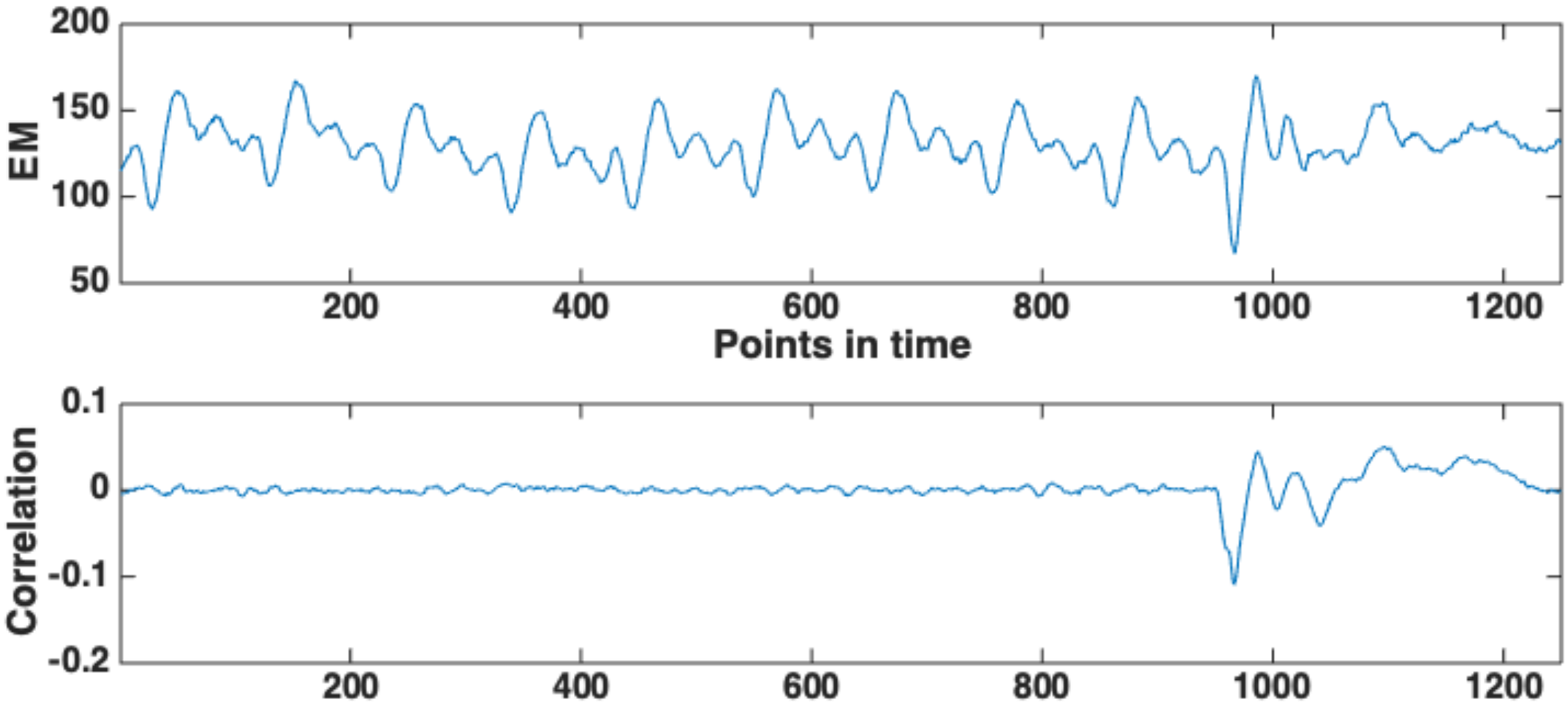
Traces

- Complete trace length: 1250
- Trace length regarding one S-box operation: approx 150



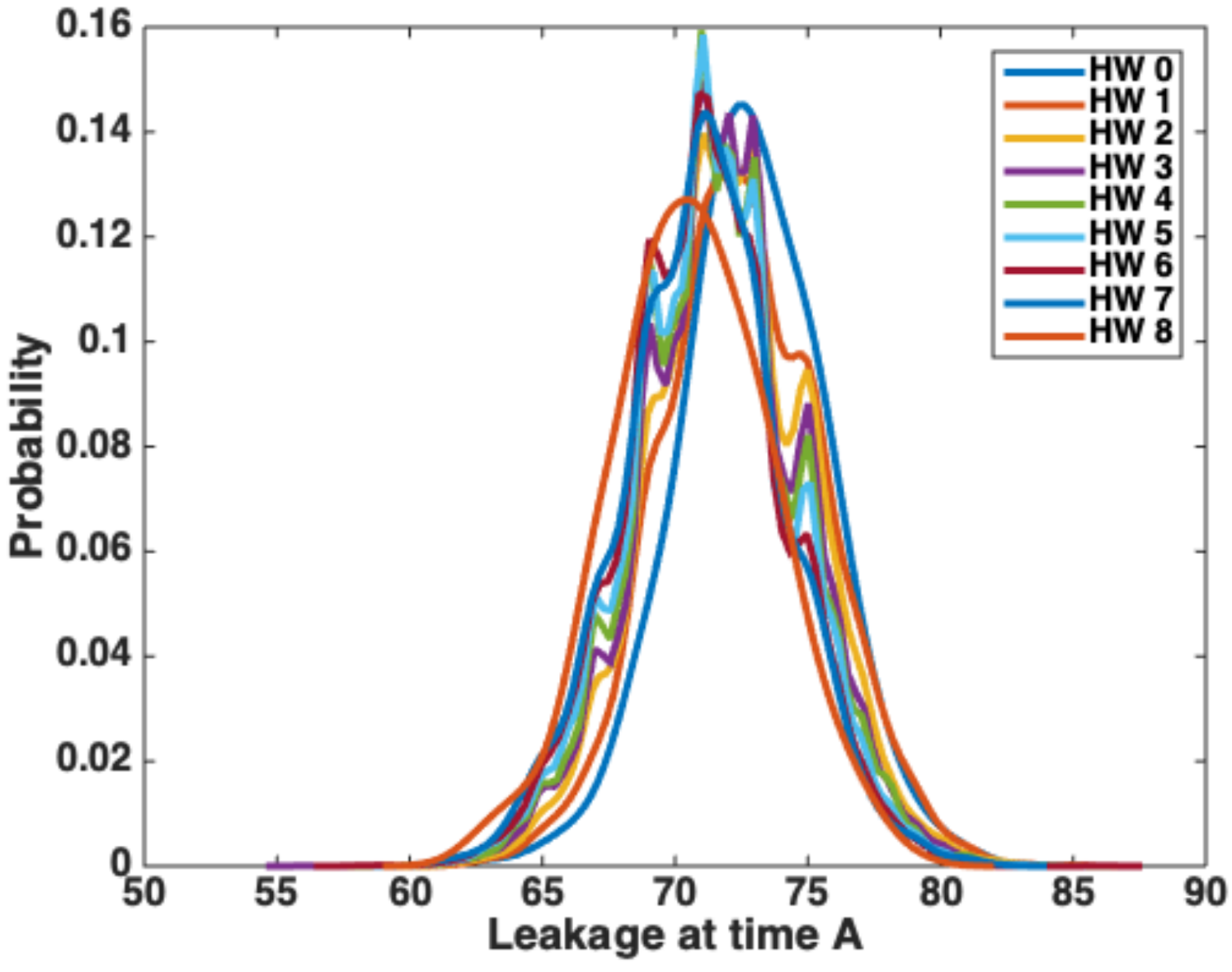
Leakage

- Correlation between HD of the Sbox output (last round) and traces



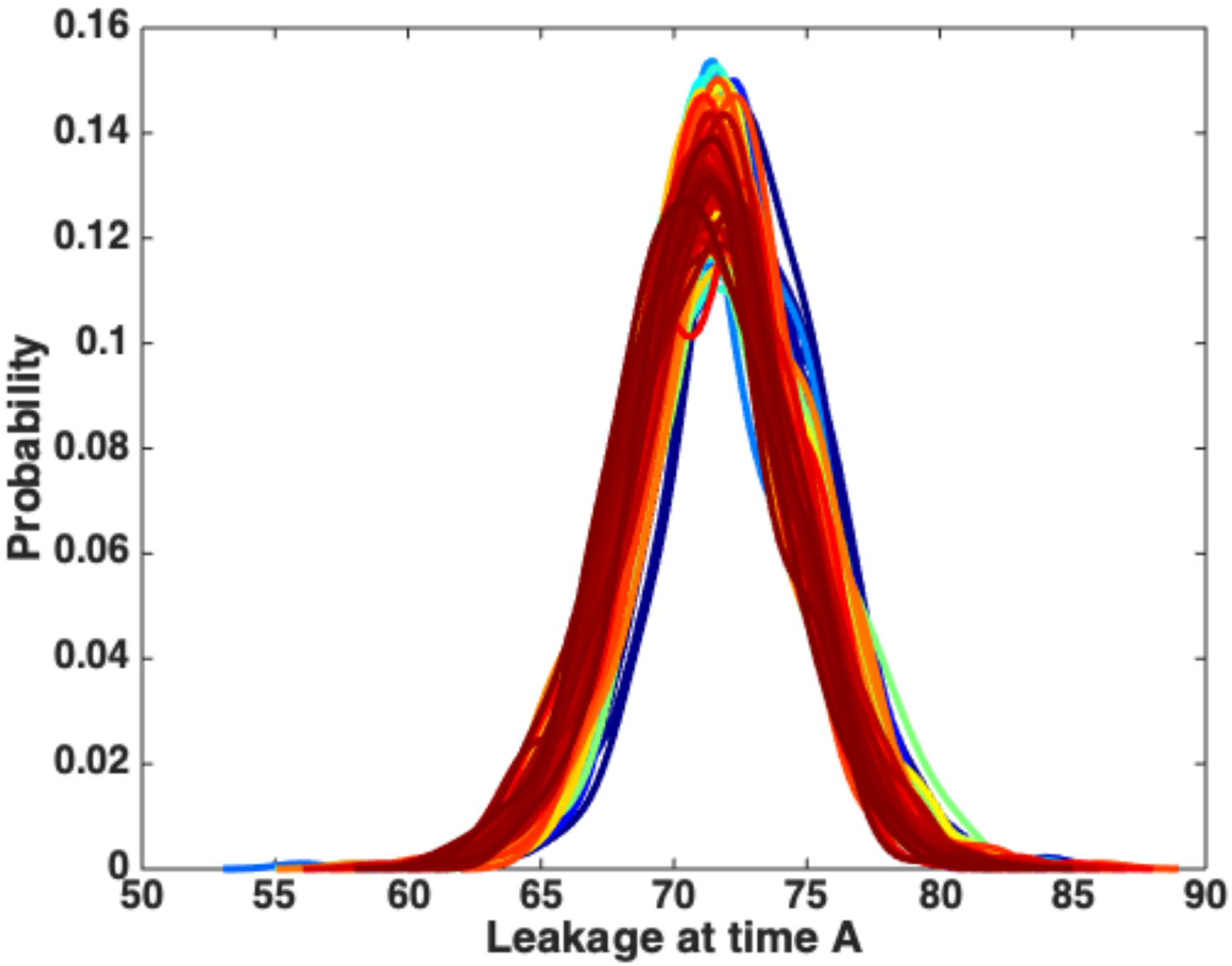
Observable leakage

- High noise scenario: densities of HWs



Observable leakage

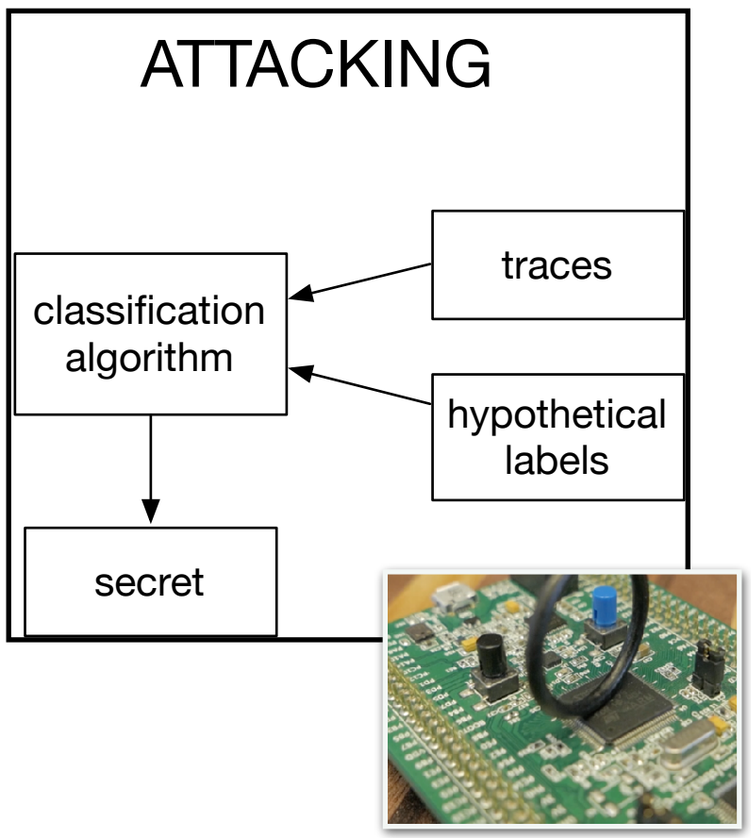
- High noise scenario: 256 classes



Attacker models

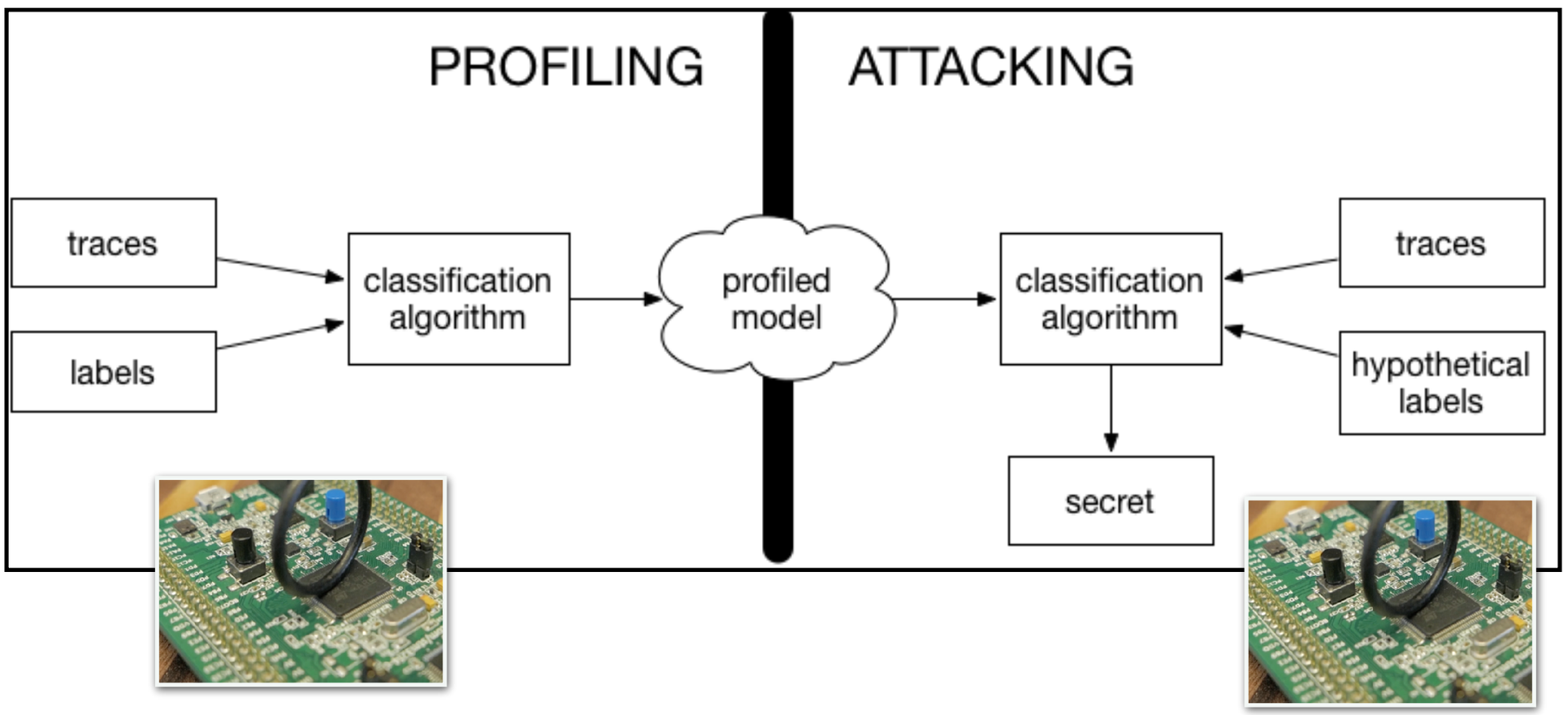
- un-profiled:
  attacker only has access to the device under attack

- weakest attacker, but more “robust"



Attacker models

- profiled (traditional view):
  attacker processes two devices - profiling and attacking

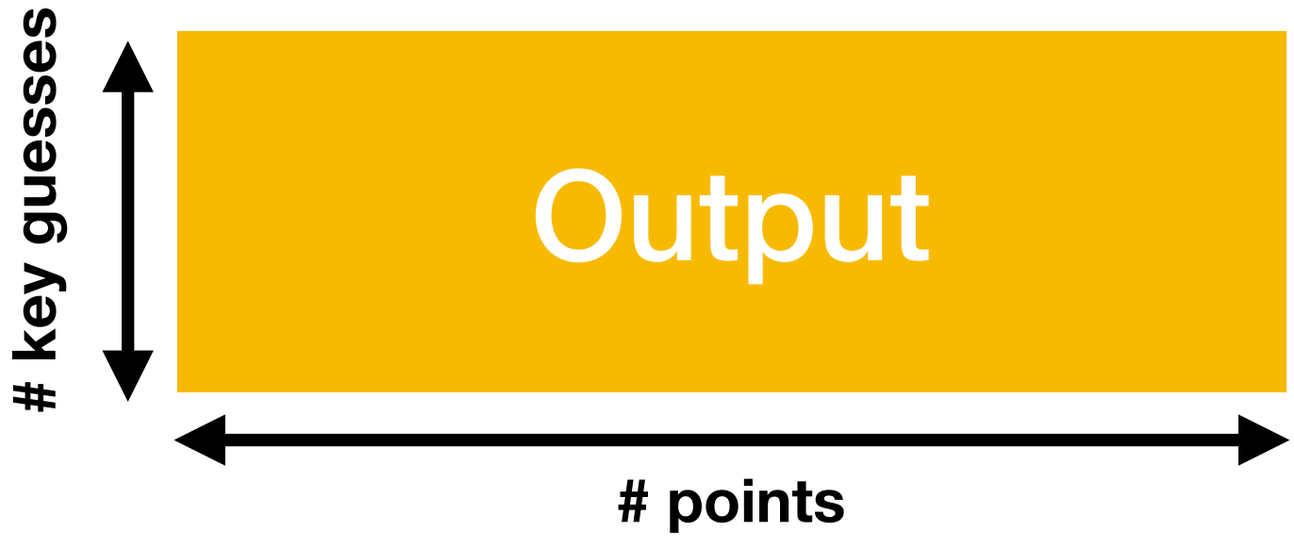


- stronger attacker, but with more pitfalls…

Side-channel attacks

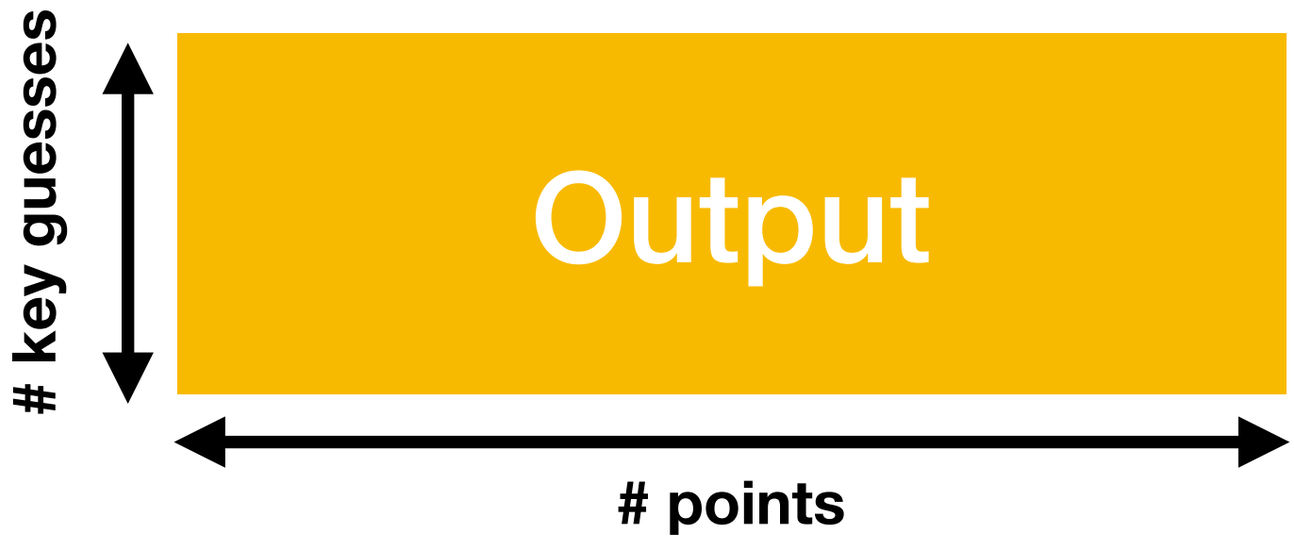
- Unprofiled:
  - Difference-of-means
  - Correlation Power Analysis (CPA)
  - Linear regression Analysis
  - Deep learning techniques (supervised)
- Profiled:
  - Template attack
  - Stochastic approach
  - Machine learning techniques
  - Deep learning techniques

Unprofiled SCA



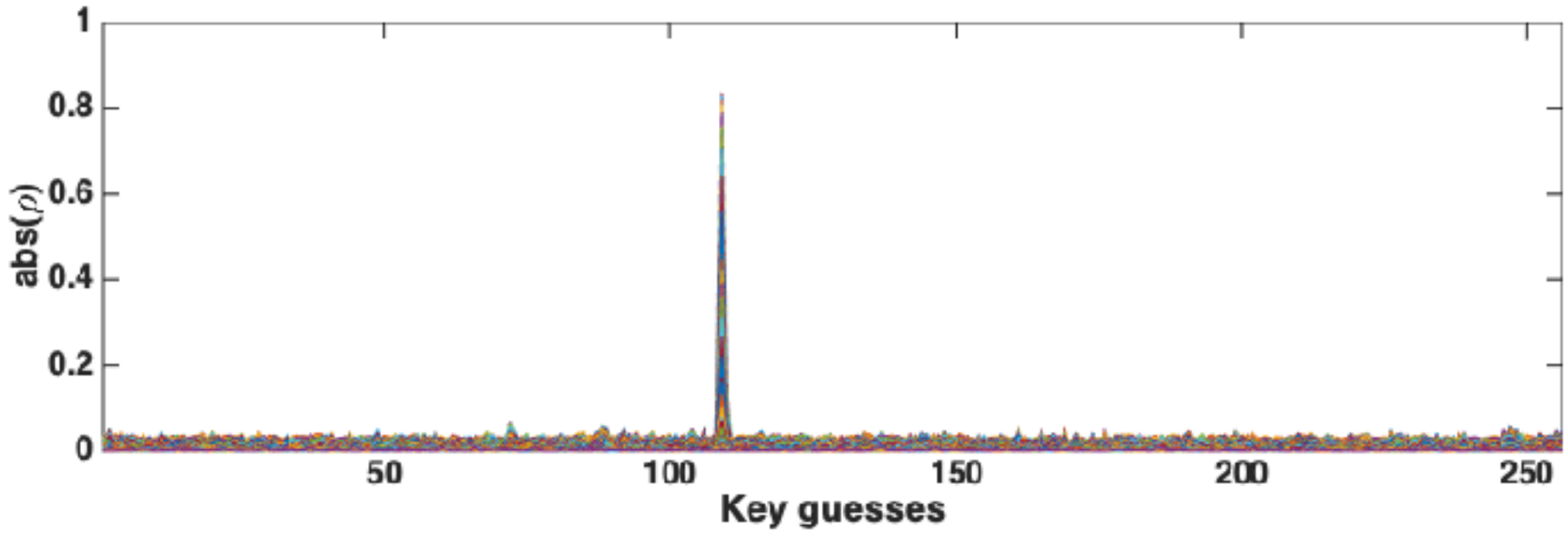
CPA

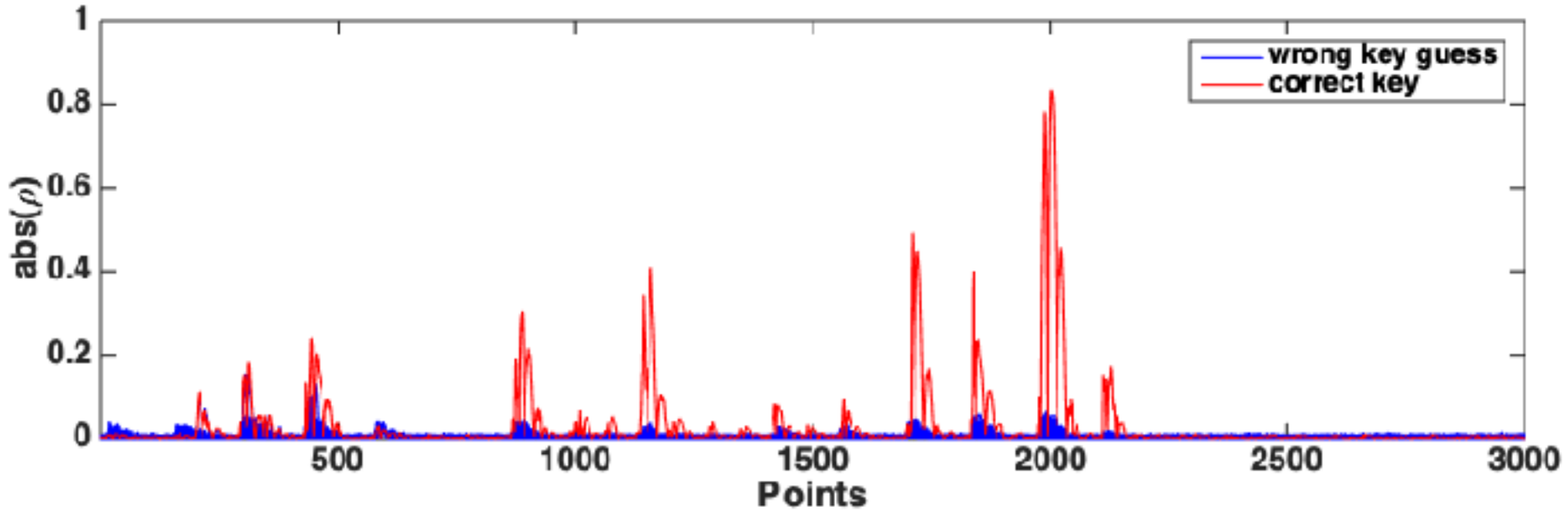
$$\rho($$ Traces , Labels $$)$$



CPA

- Dataset 1: Labels = output of the S-Box in the first round



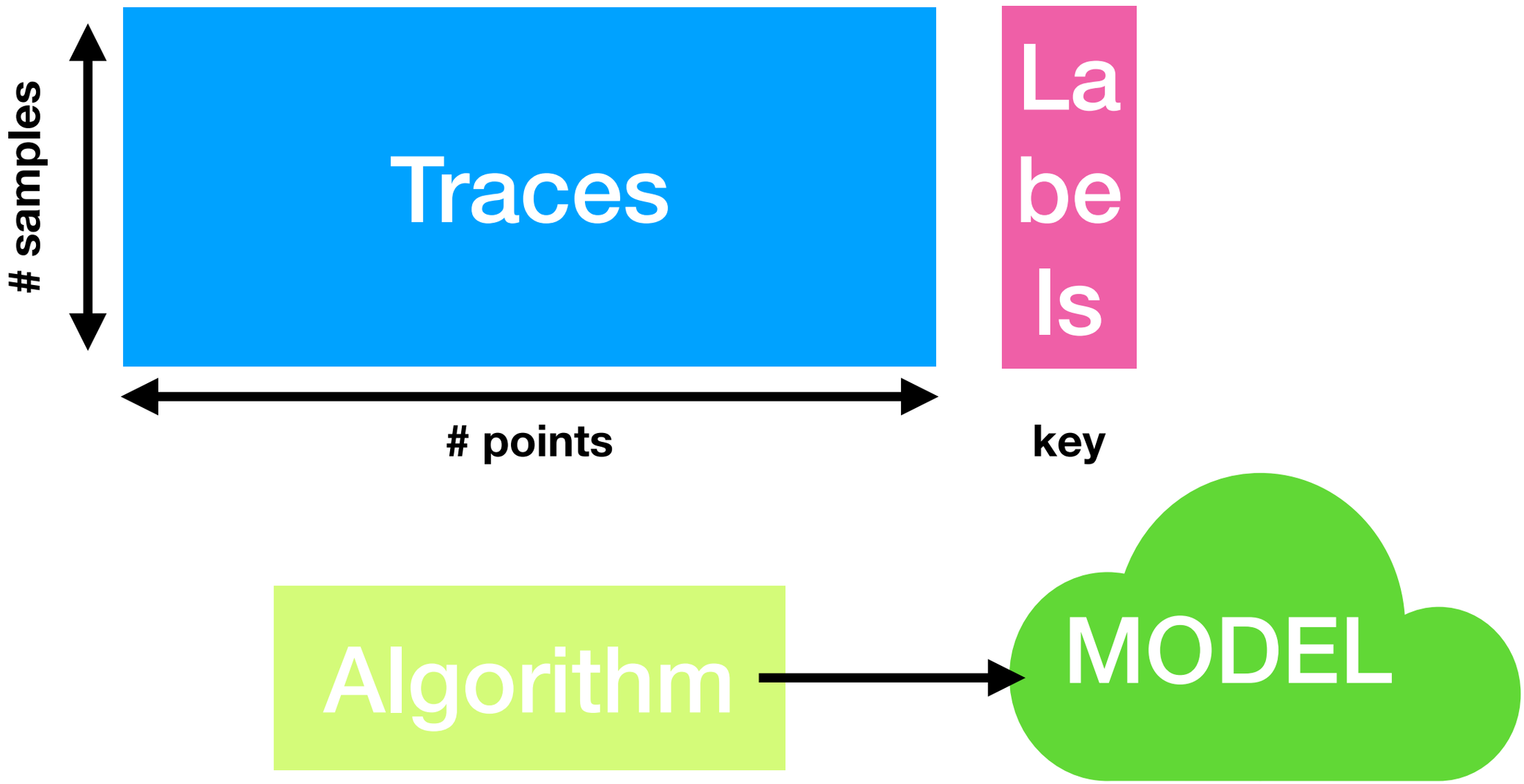


Profiled side-channel

- Profiling phase:
  - classification (Template attack, SVM, RF, Deep learning)
  - regression (Stochastic approach)
- Attacking phase:
  - maximum likelihood principle

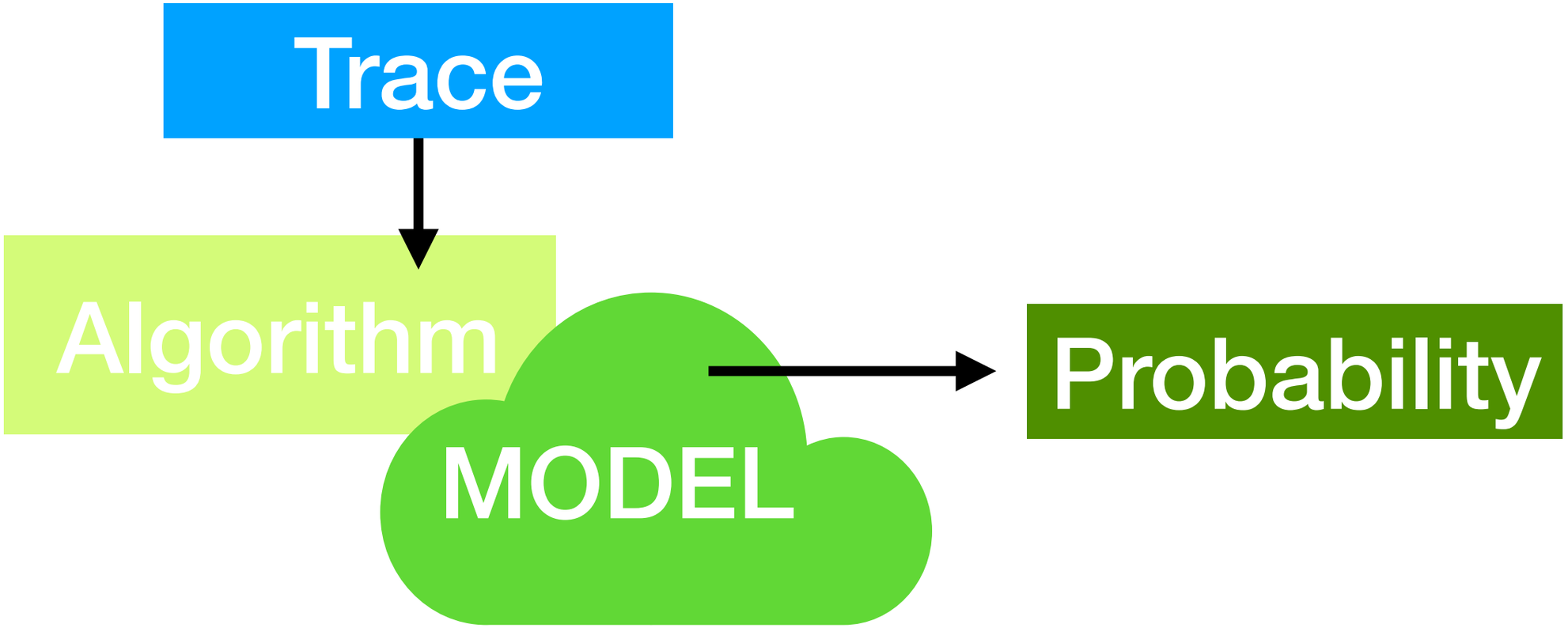
Profiled SCA

- Profiling phase: building model



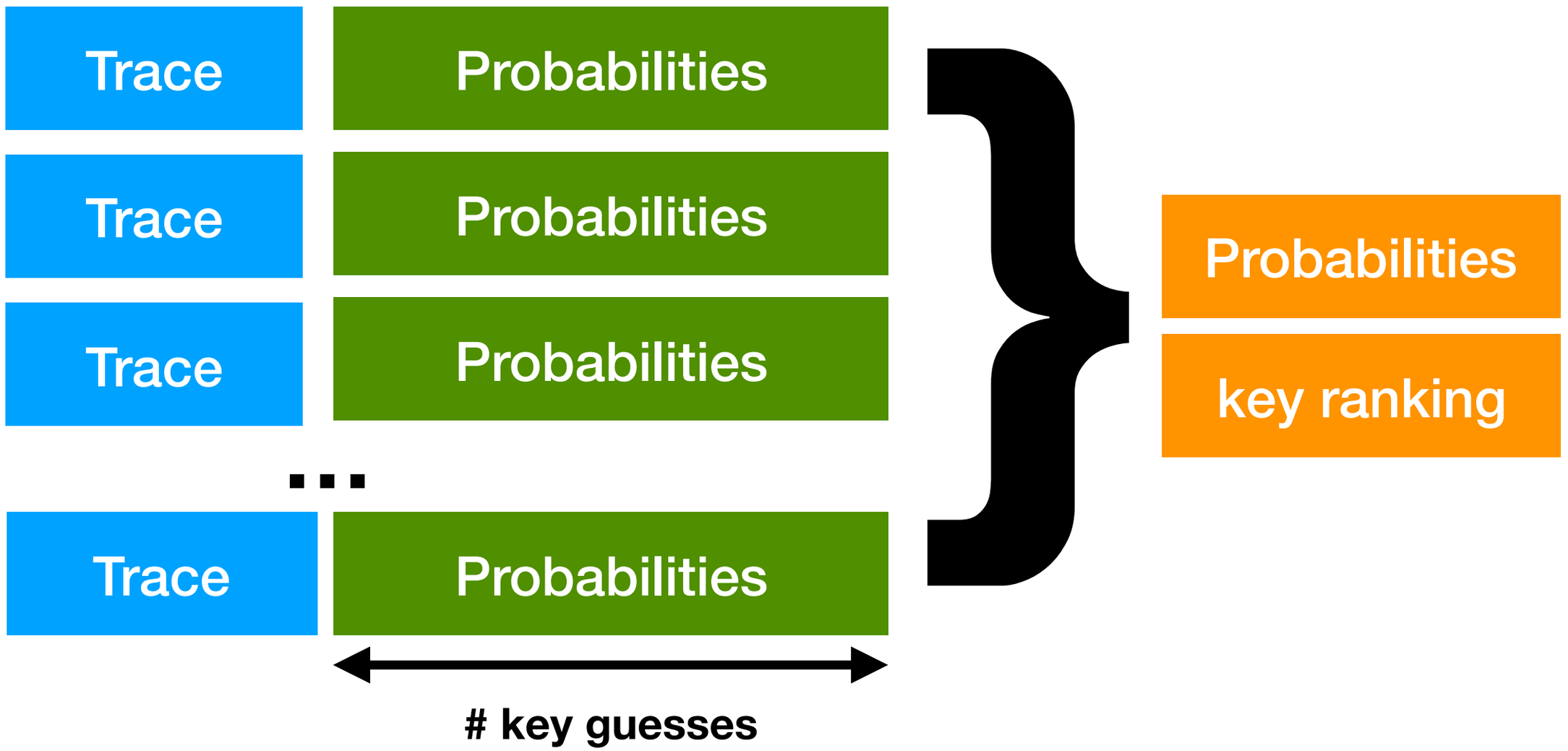
Profiled SCA

- For each trace in the attacking phase, get the probability that the trace belongs to a certain class label



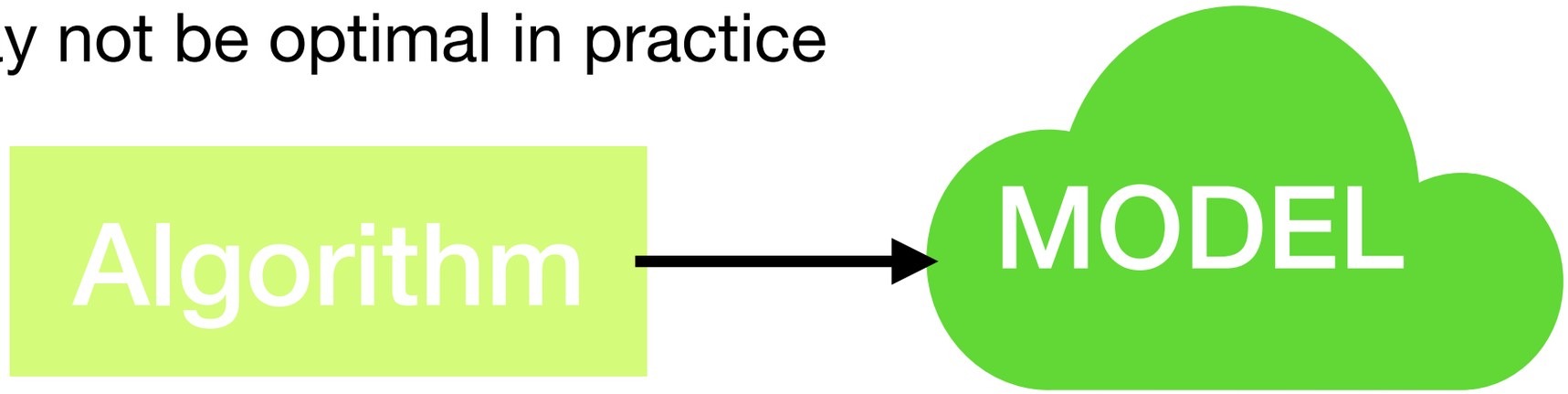
Profiled SCA

- Maximum likelihood principle to calculate that a set of traces belongs to a certain key



Template attack

- first profiled attack
- optimal from an information theoretical point of view
- may not be optimal in practice

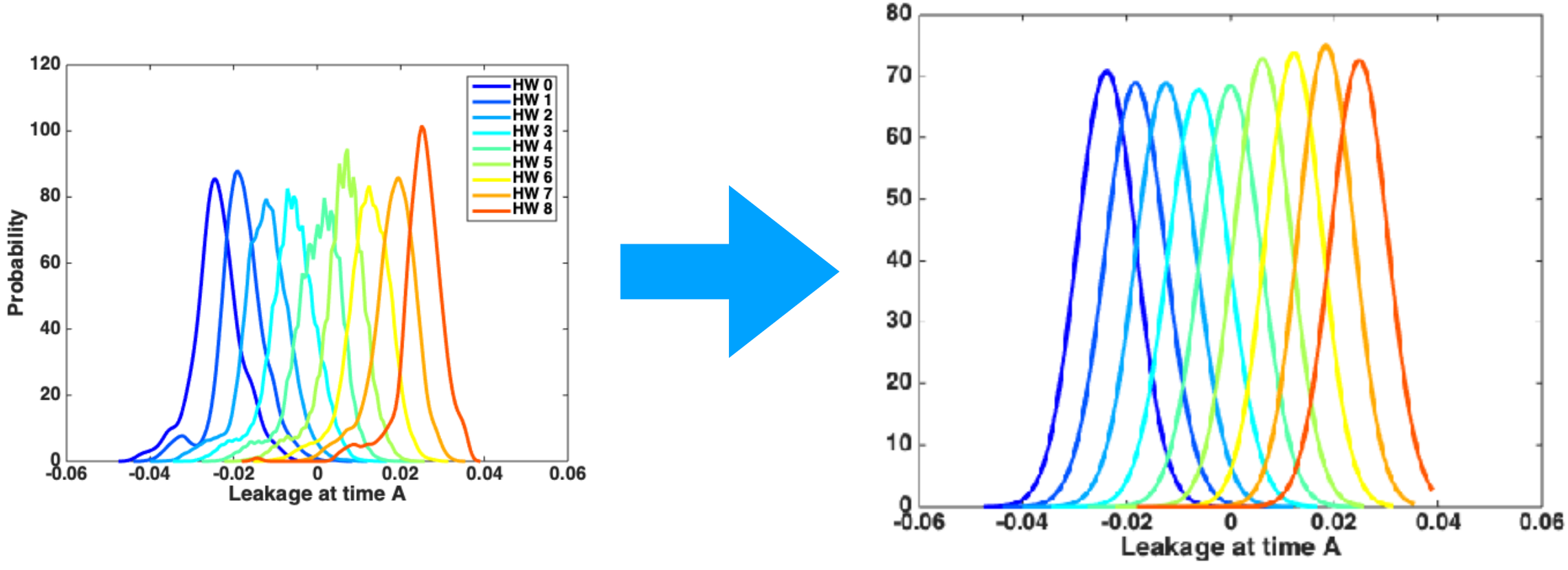


Density estimation **densities**

- often works with the pre-assumption that the noise is normal distributed
- advantage of being easier to estimate:
  mean and covariances for each class label
- pooled version

Template attack

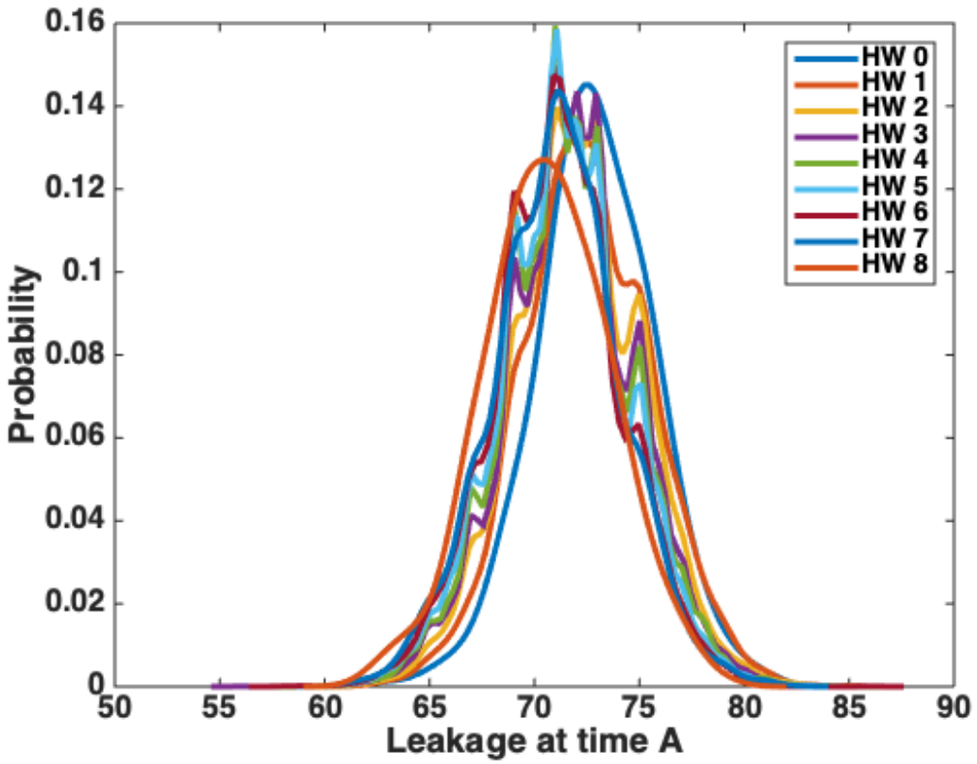
- Dataset 1: low noise
- Assumption of normal distribution

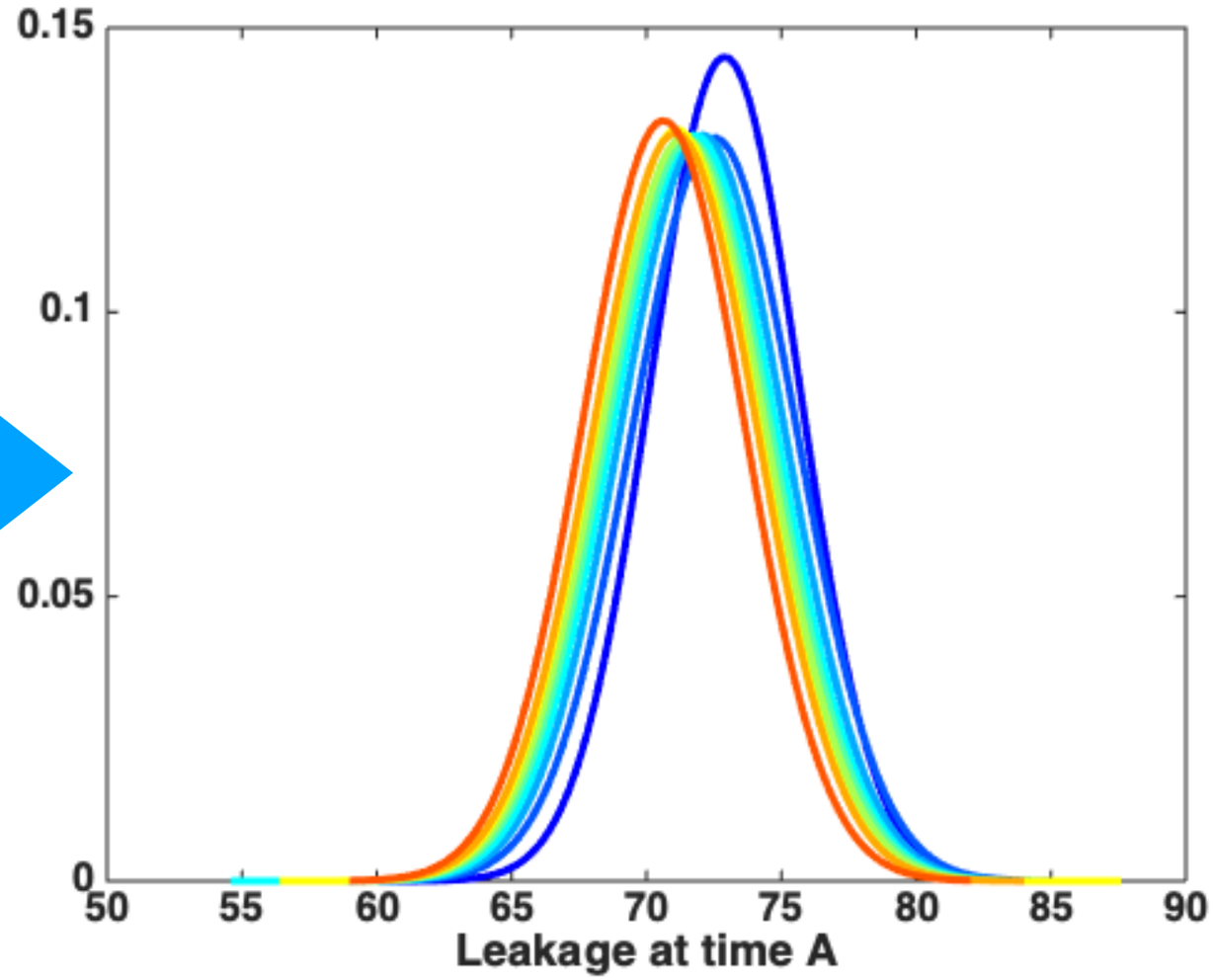


multivariate: means and *covariances* over a set of points

Template attack

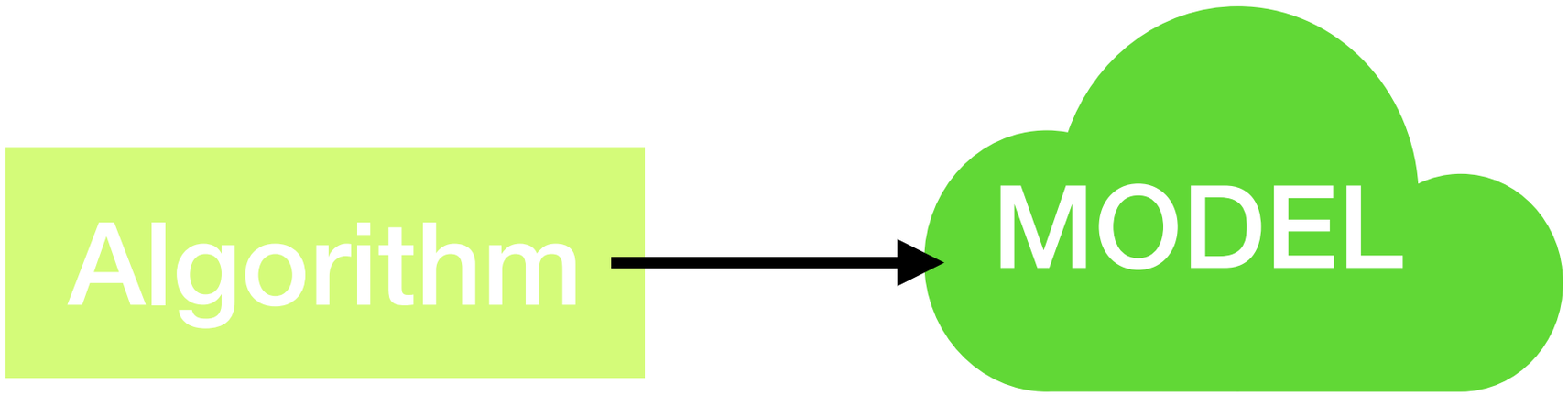
- Dataset 2: high noise





Stochastic Approach

- uses regression instead of classification
- estimate a function that models the leakage
- constructive: may provide detailed feedback about leakage “source”



Linear regression

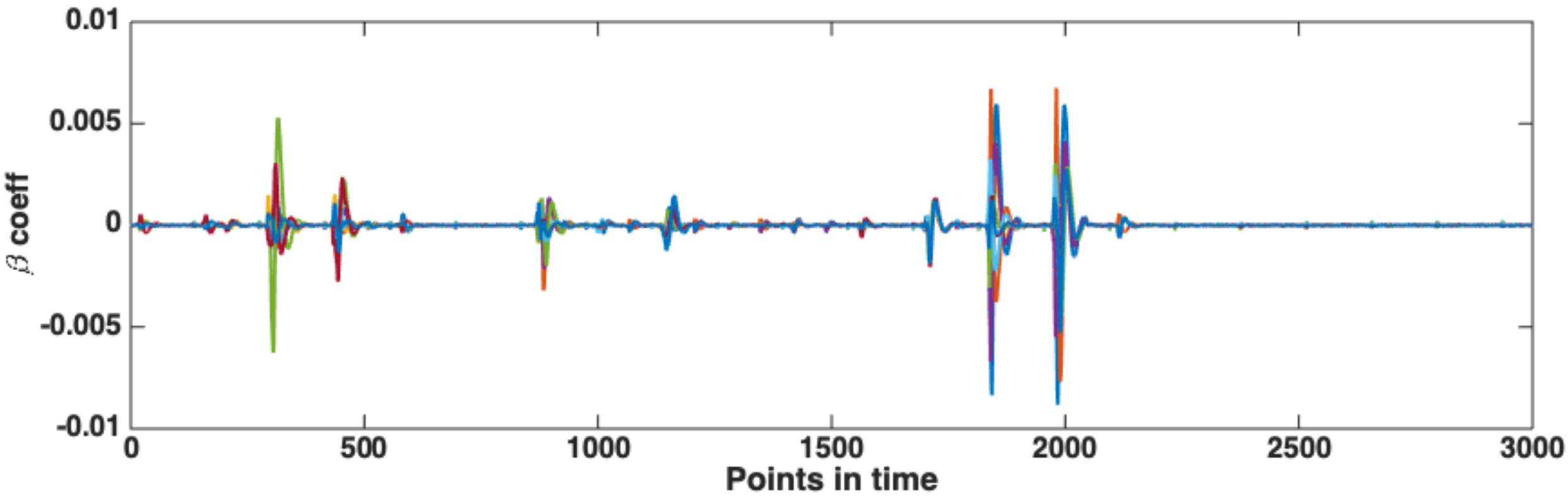
regression coefficients/ beta-coefficients

Stochastic Approach

- Regressors/ “basis (functions)” for linear regression:
  - 9-dimensional basis: const + bits
  - 37-dimensional basis: const + bits + prod 2 bits
  - 92-dimensional basis: const + bits + prod 2 bits + prod 3 bits
  - …
  - 256-dimensional basis: const + bits + prod 2 bits + prod 3 bits + prod 4 bit + prod 5 bit + prod 6 bit + … + prod 8 bit

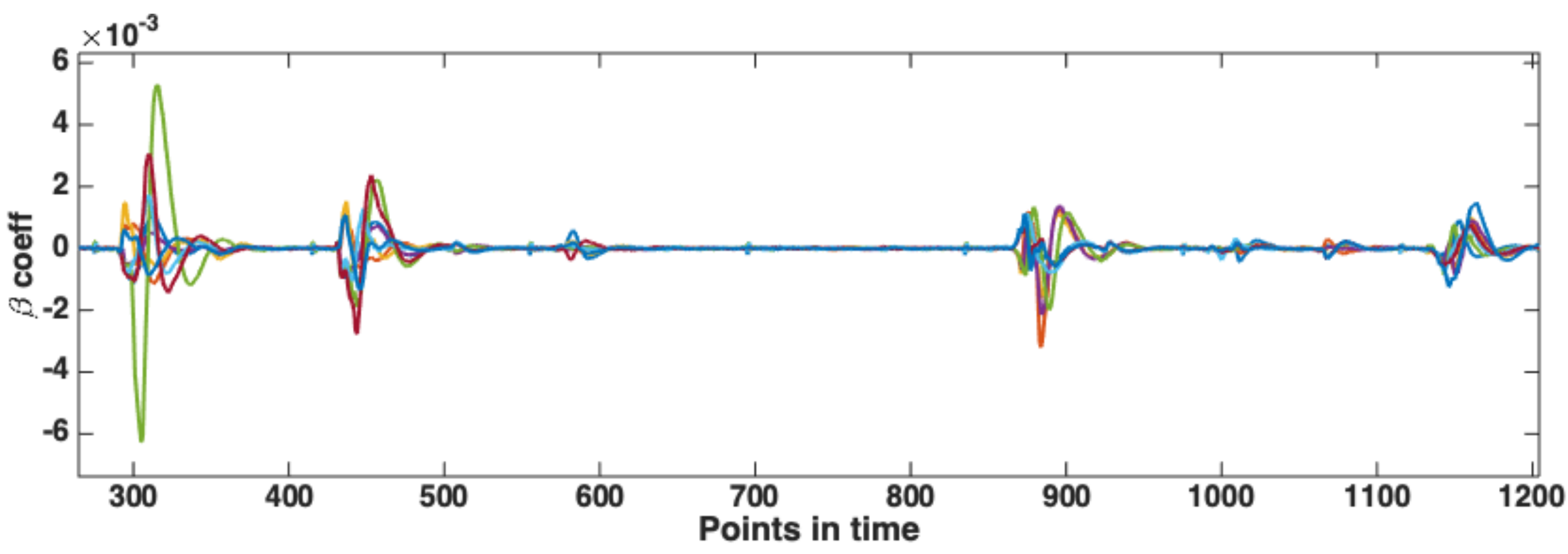
Stochastic Approach

- Dataset 1: low noise
- Basis: 9-dimensional



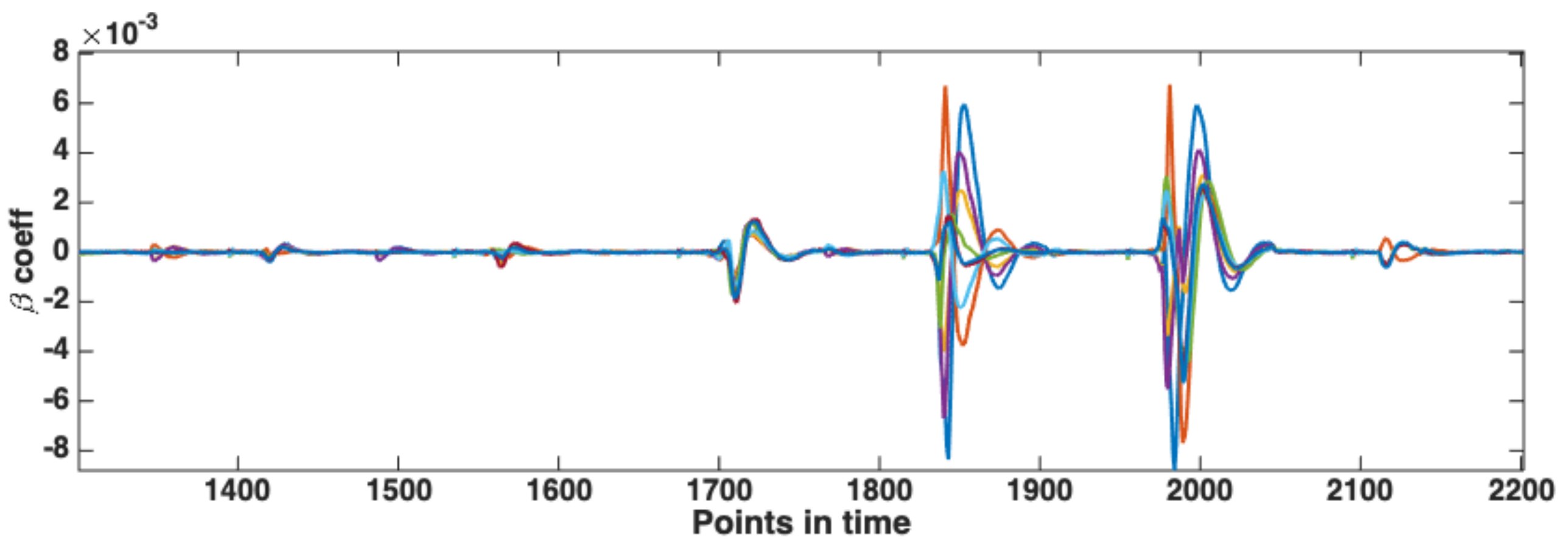
Stochastic Approach

- Dataset 1: low noise
- 9-dim basis, zoom in



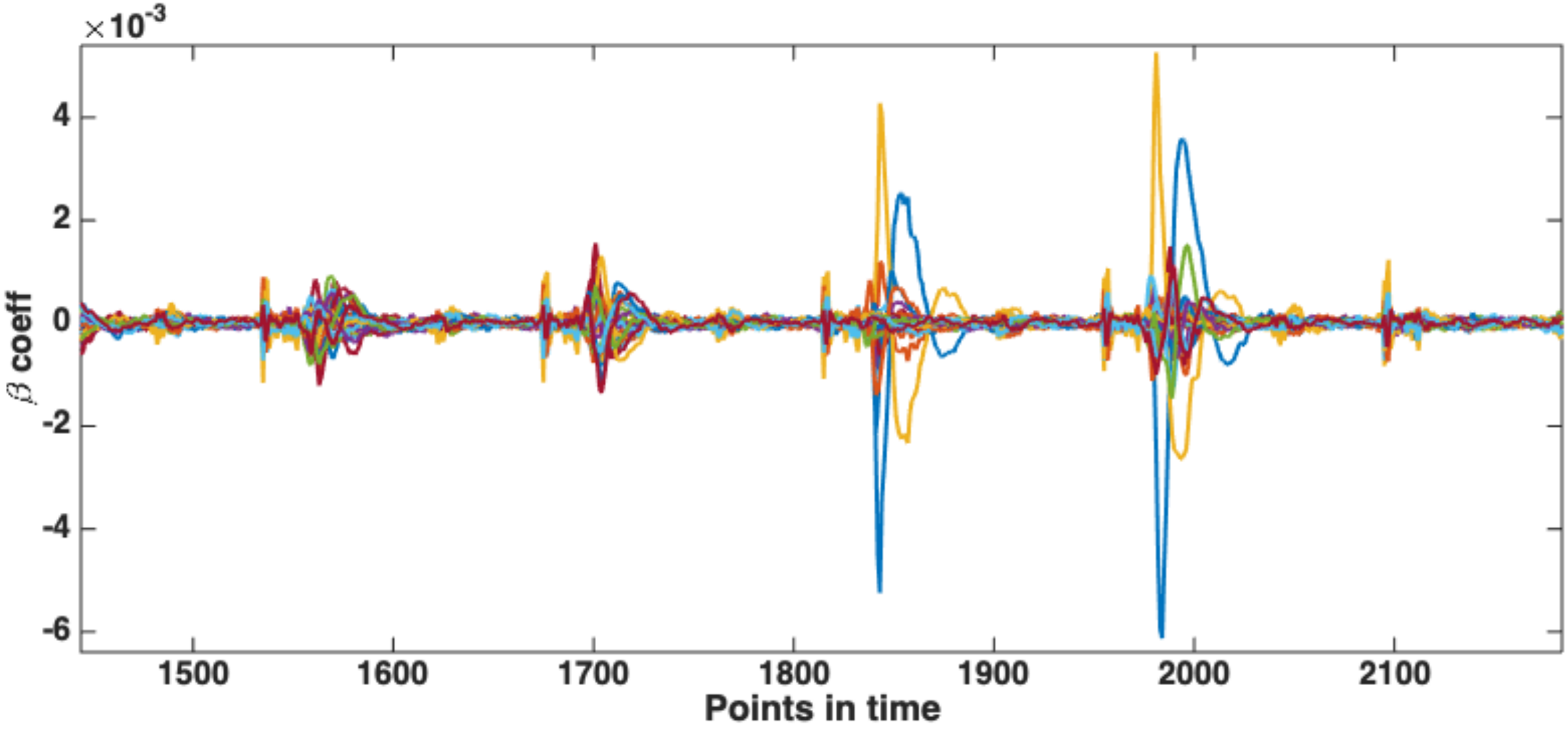
Stochastic Approach

- Dataset 1: low noise
- 9-dim basis, zoom in



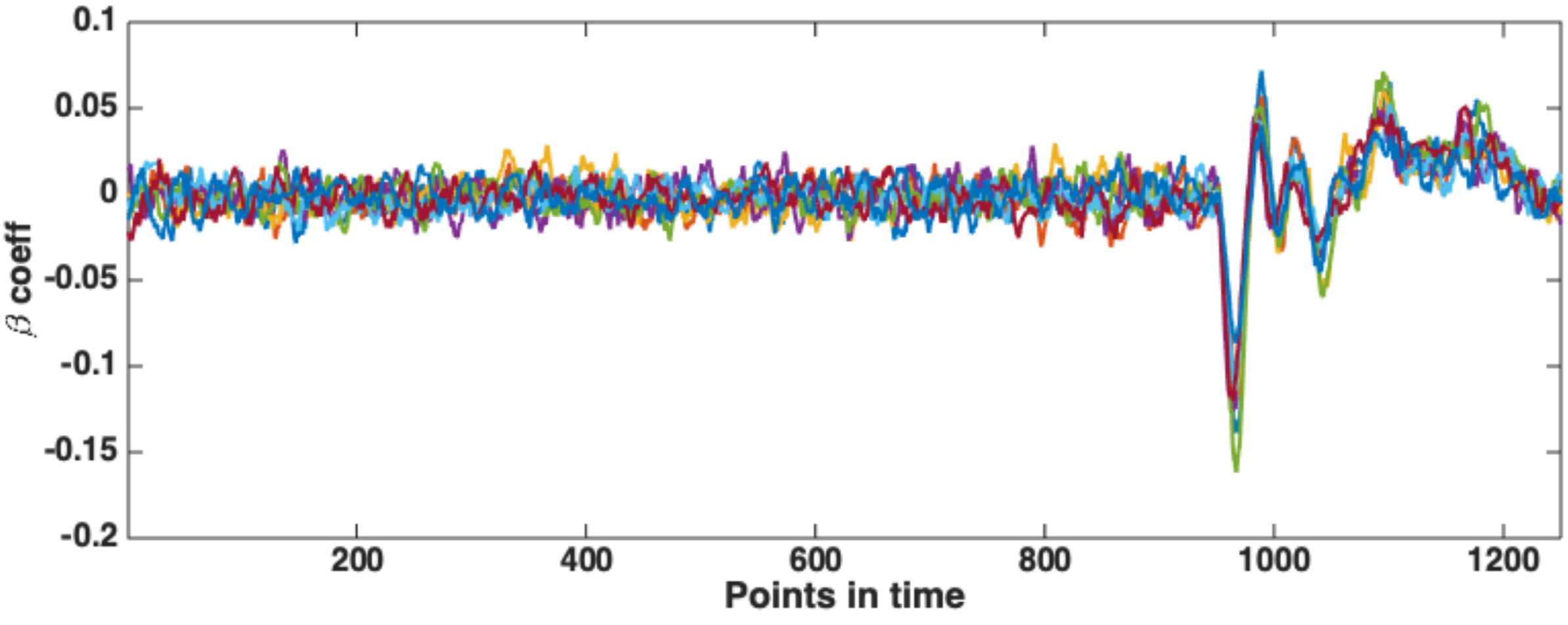
Stochastic Approach

- Dataset 1: low noise
- 37-dim basis, zoom in



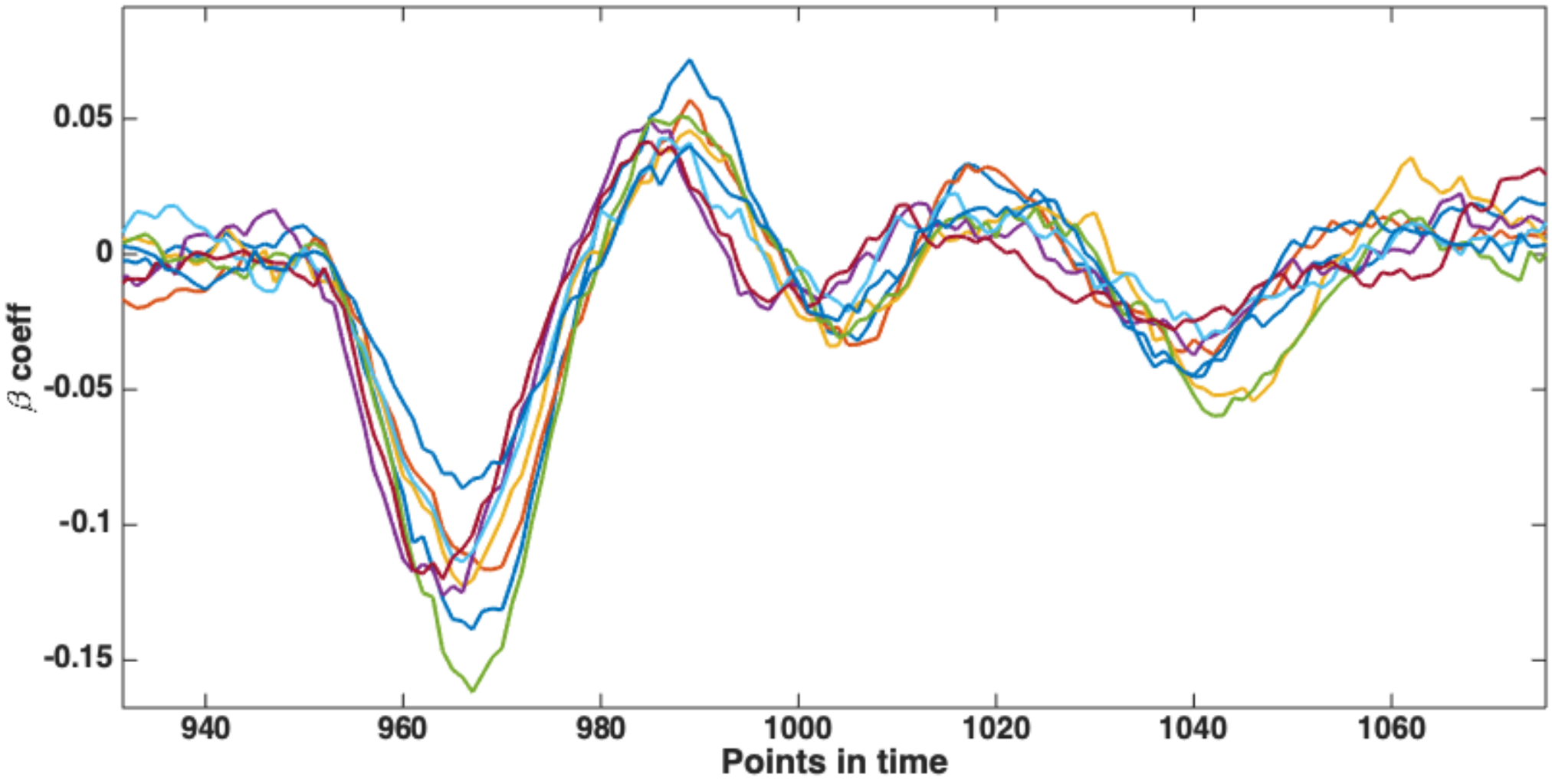
Stochastic Approach

- Dataset 2: high noise

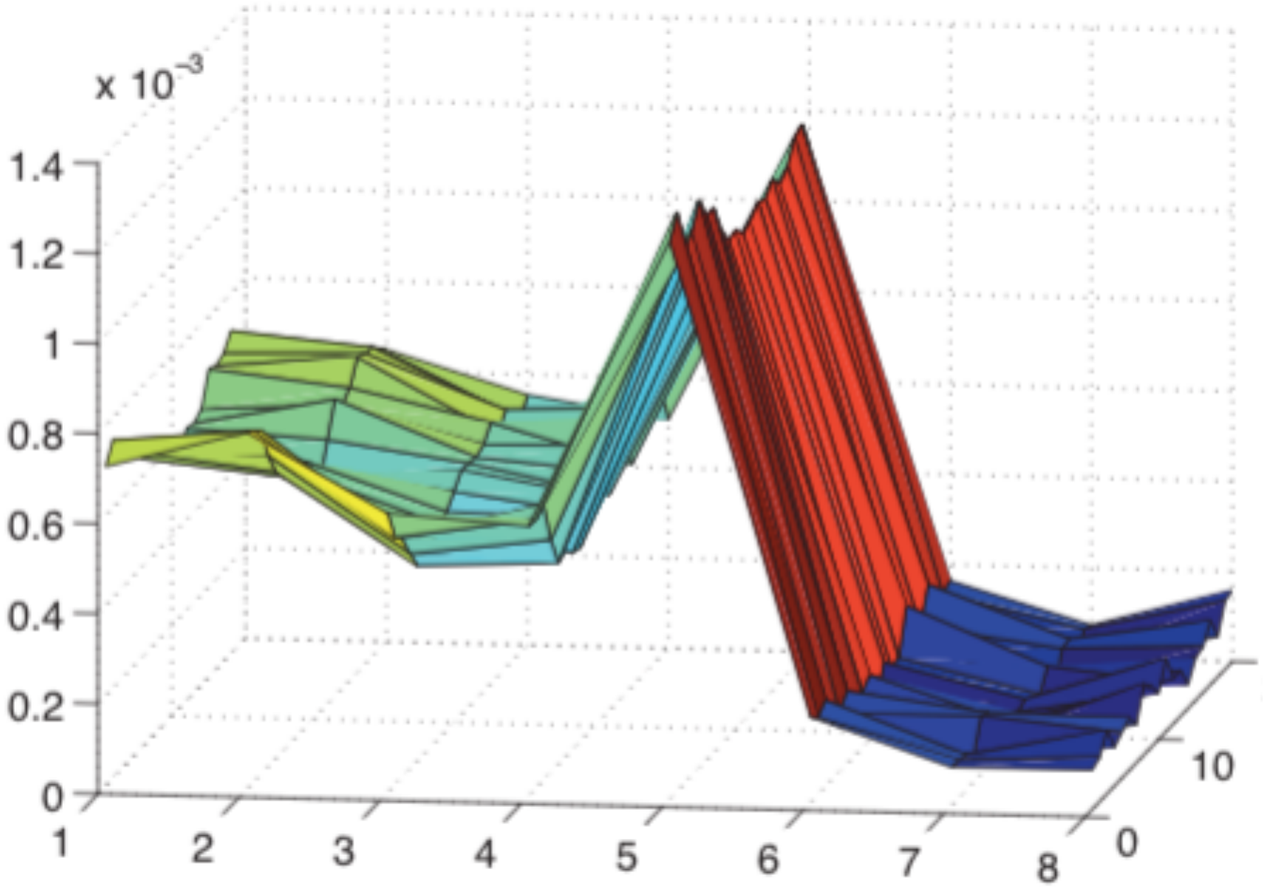


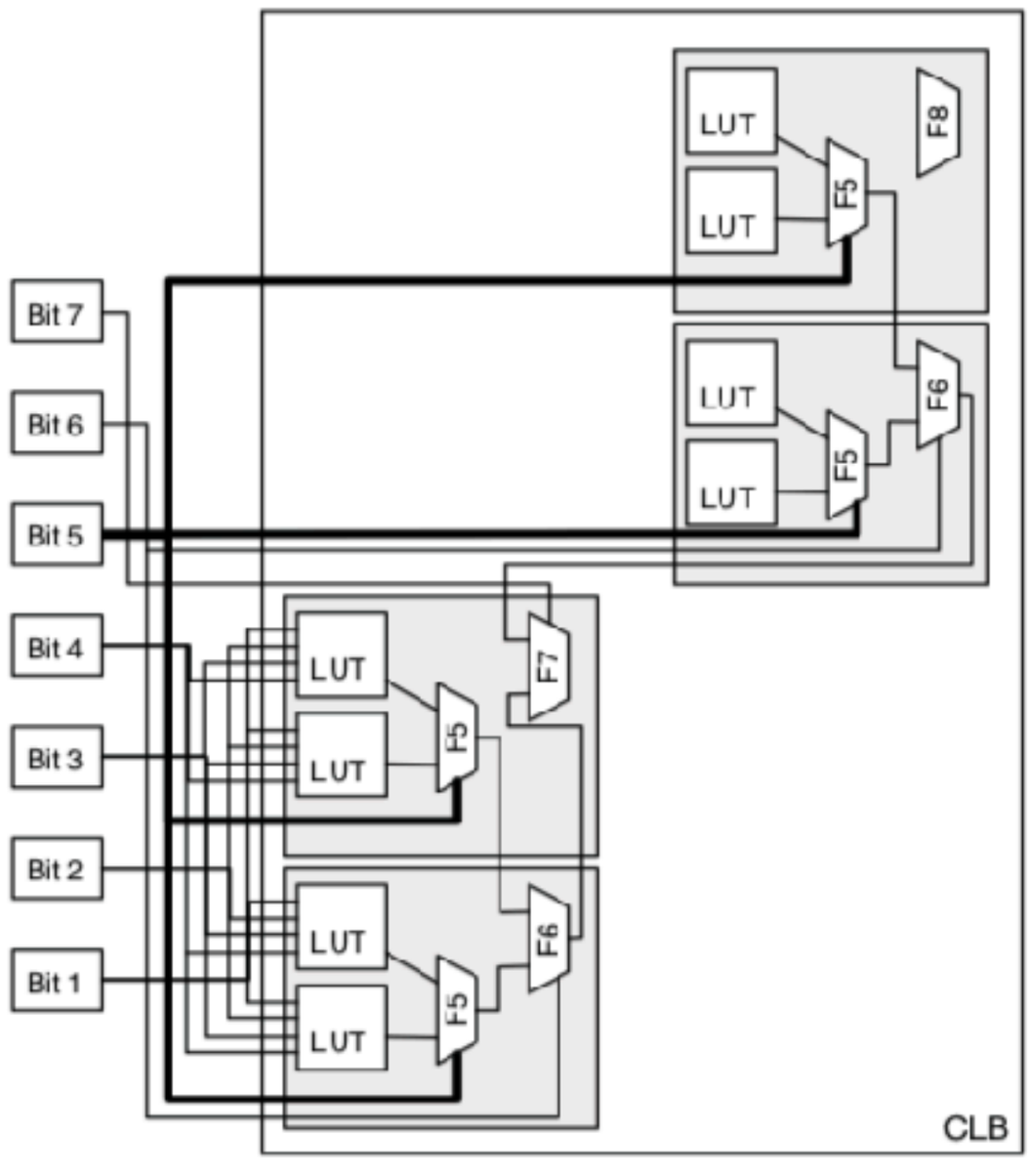
Stochastic Approach

- Dataset 2: high noise
- zoom in



Constructiveness?





Michael Kasper, Werner Schindler, Marc Stöttinger:
A stochastic method for security evaluation of cryptographic FPGA implementations. FPT 2010: 146-153

Success rate

- Success rate: average estimated probability of success
- empirically: using measurements/ simulations
- theoretically: using closed-form expressions

- For CPA and template attack the theoretical success rate depends on 3 factors
  - number of measurements
  - signal-to-noise ratio
  - *confusion coefficient*

Yunsi Fei, Qiasi Luo, A. Adam Ding:
A Statistical Model for DPA with Novel Algorithmic Confusion Analysis. CHES 2012: 233-250

Confusion Coefficient

- Interestingly, predictions for different key guesses are not independent

$$Y(k) = \mathrm{HW}(\mathtt{Sbox}[T \oplus k])$$

- Confusion coefficient describes the relationship

$$\kappa(k^*, k) = \mathbb{E}\left\{\left(\frac{Y(k^*) - Y(k)}{2}\right)^2\right\}$$

- (simplified) metric: the lower the minimum confusion coefficient (over all keys) the higher the side-channel resistance

Sylvain Guilley, Annelie Heuser, Olivier Rioul:
A Key to Success - Success Exponents for Side-Channel Distinguishers. INDOCRYPT 2015: 270-290

SBoxes

- SBoxes with optimal cryptographic properties

4-bit S-boxes

- KLEIN
- Midori (1/2)
- Mysterion
- Piccolo
- PRESENT / LED
- Pride
- PRINCE
- Rectangle
- Skinny

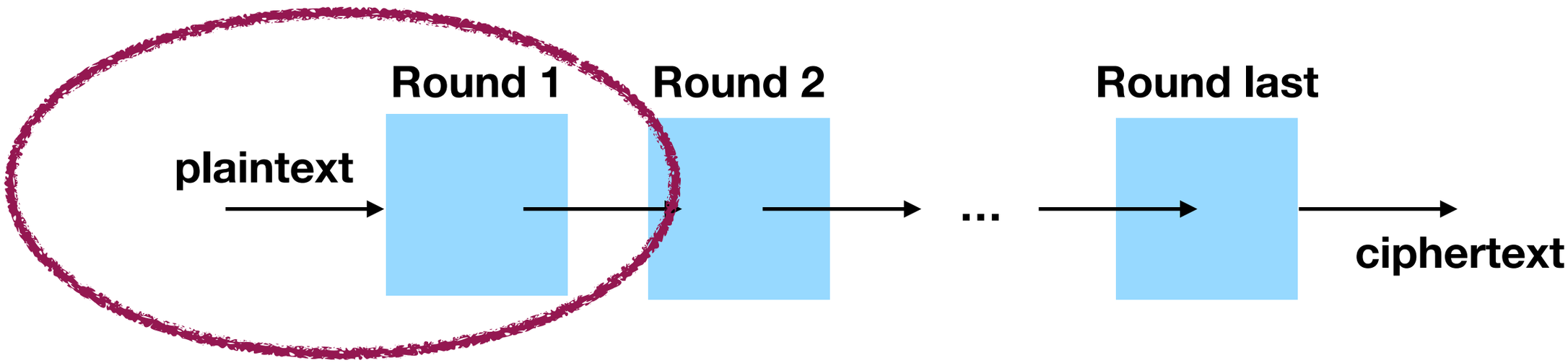
8-bit S-boxes

- AES
- Robin
- Zorro

A Heuser, S Picek, S Guilley, N Mentens Lightweight ciphers and their side-channel resilience, IEEE Transactions on Computers

Side-Channel Exploitation

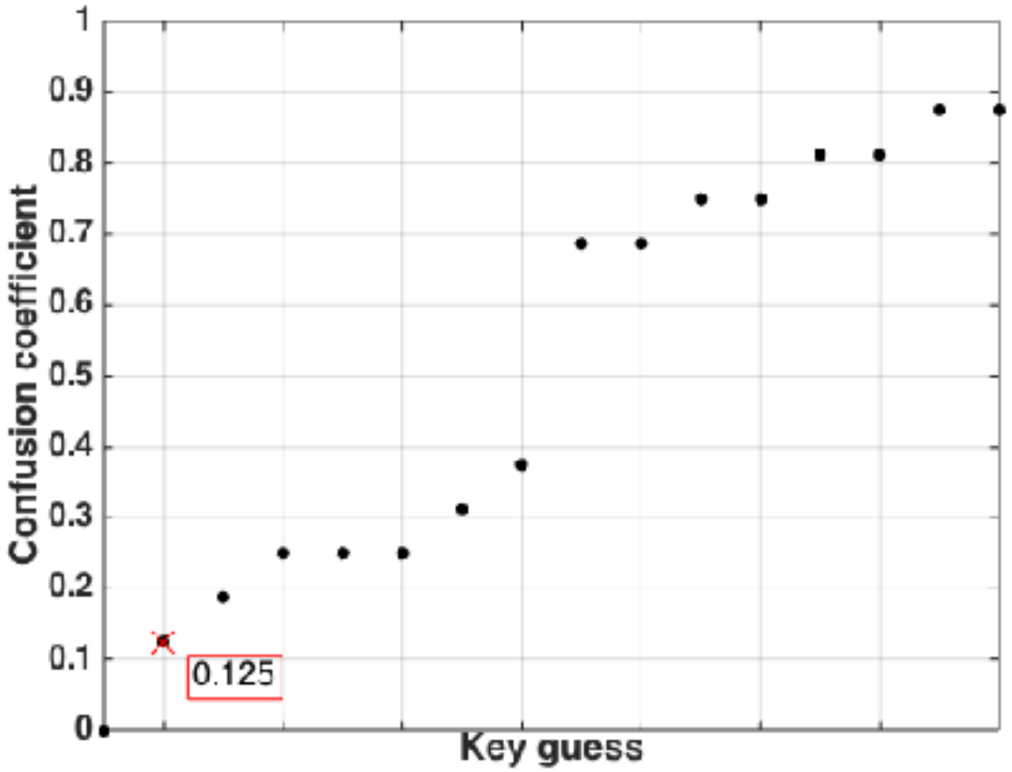
- Success depends on the confusion coefficients depending on the SBox



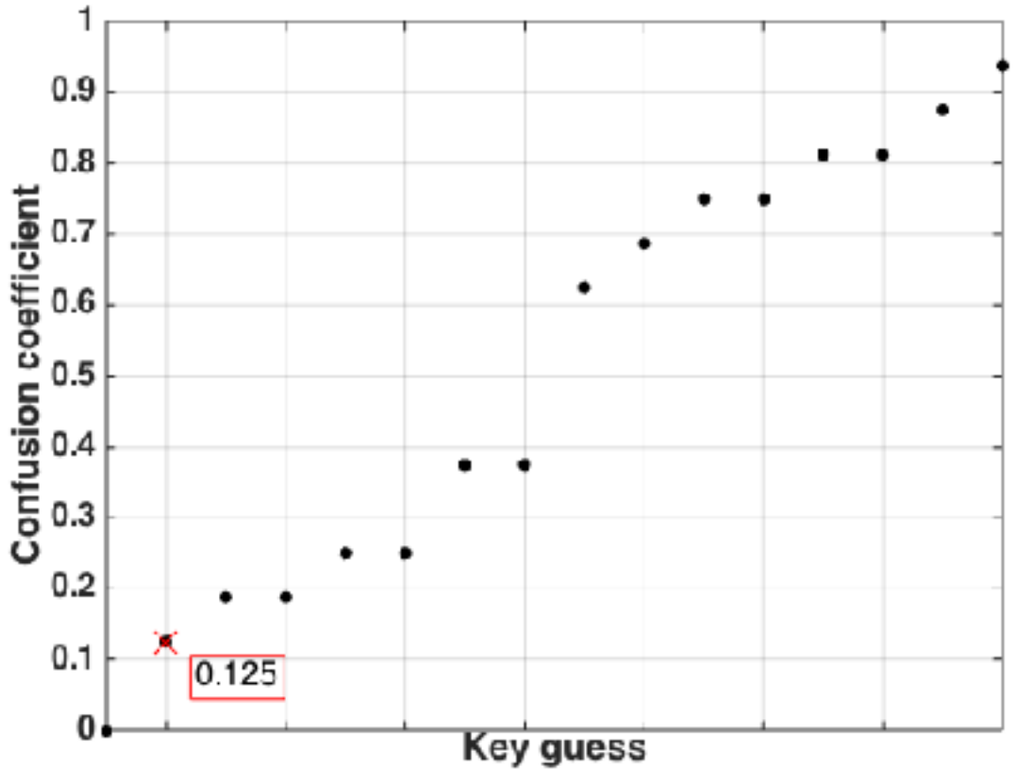
$$X = \mathrm{HW}(\mathtt{Sbox}[P \oplus k^*]) + N$$

Confusion Coefficients

the lower the minimum the higher the resistance



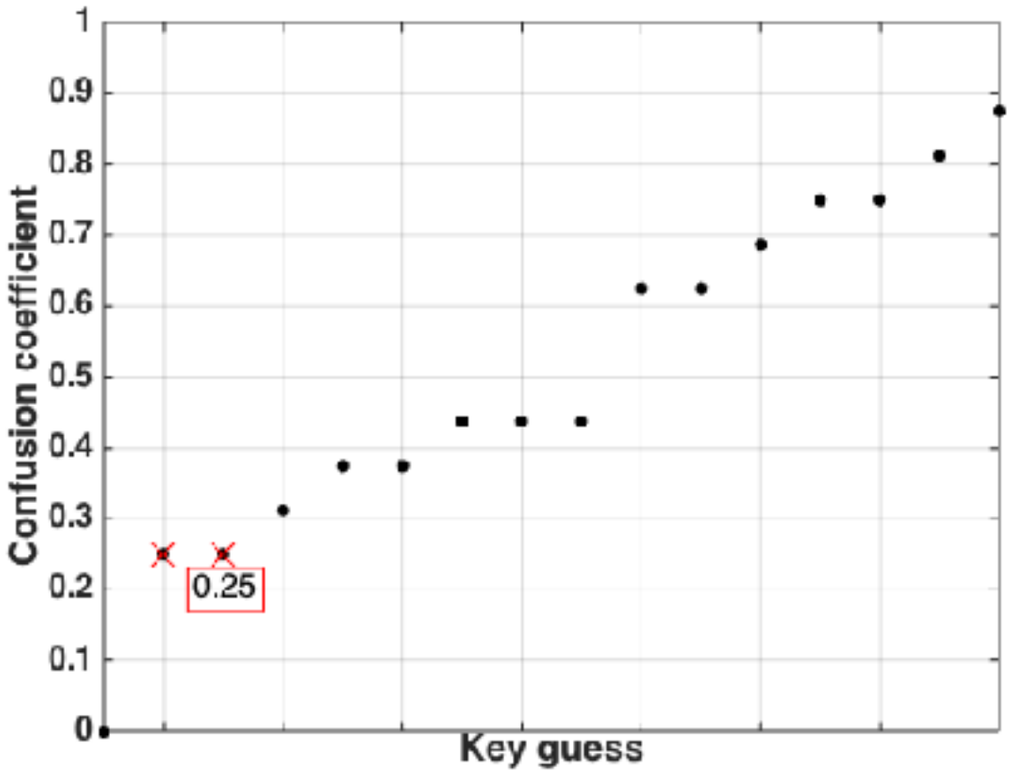
KLEIN



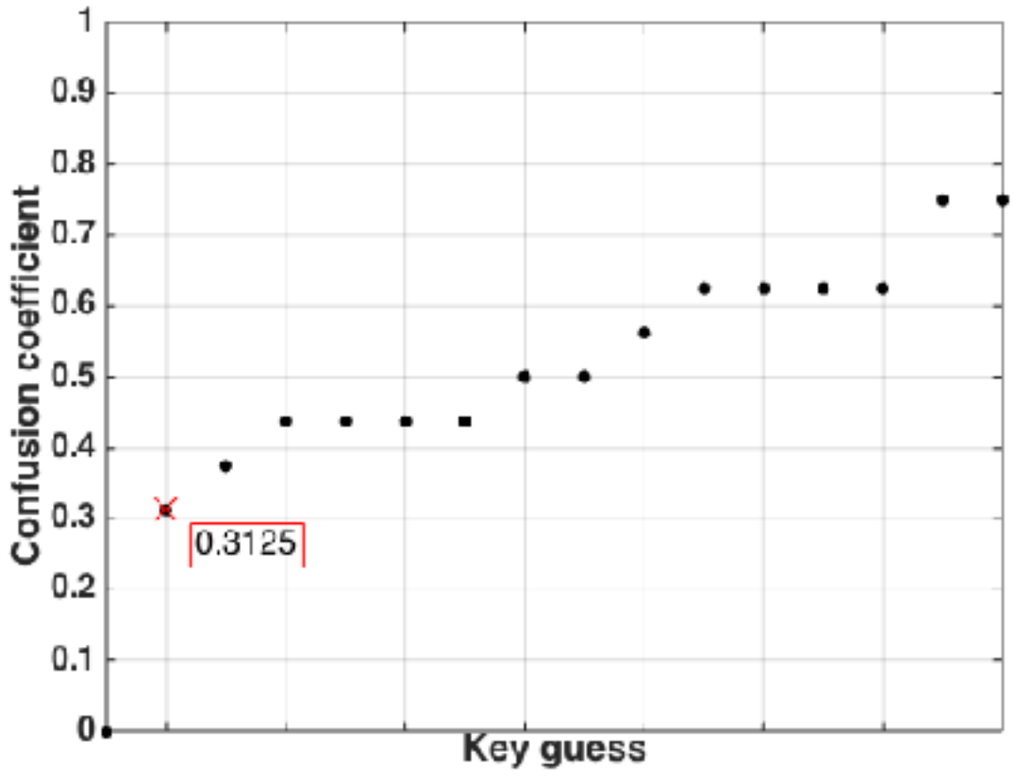
Midori 1

Confusion Coefficients

the lower the minimum the higher the resistance



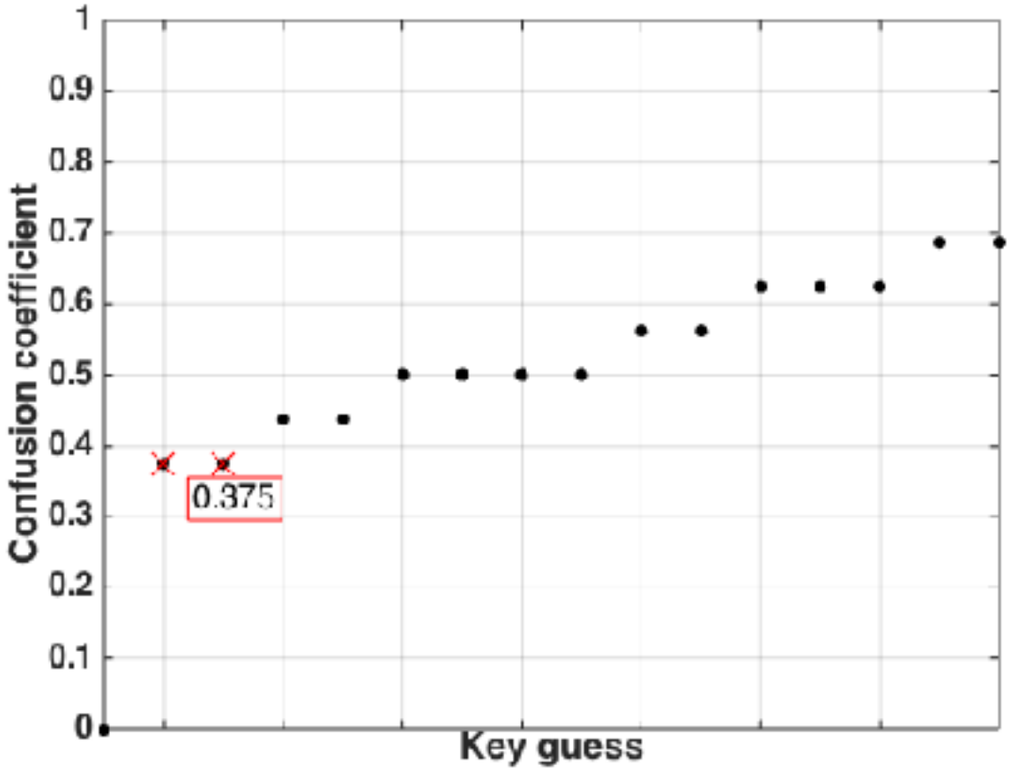
Midori 2



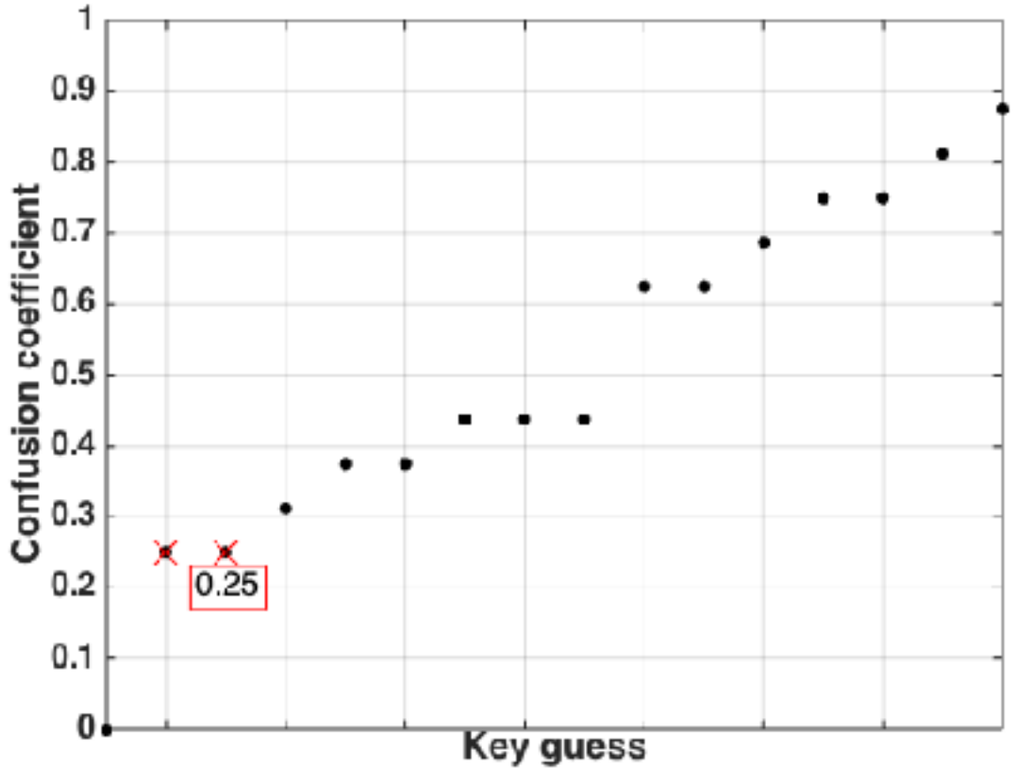
Mysterion

Confusion Coefficients

the lower the minimum the higher the resistance



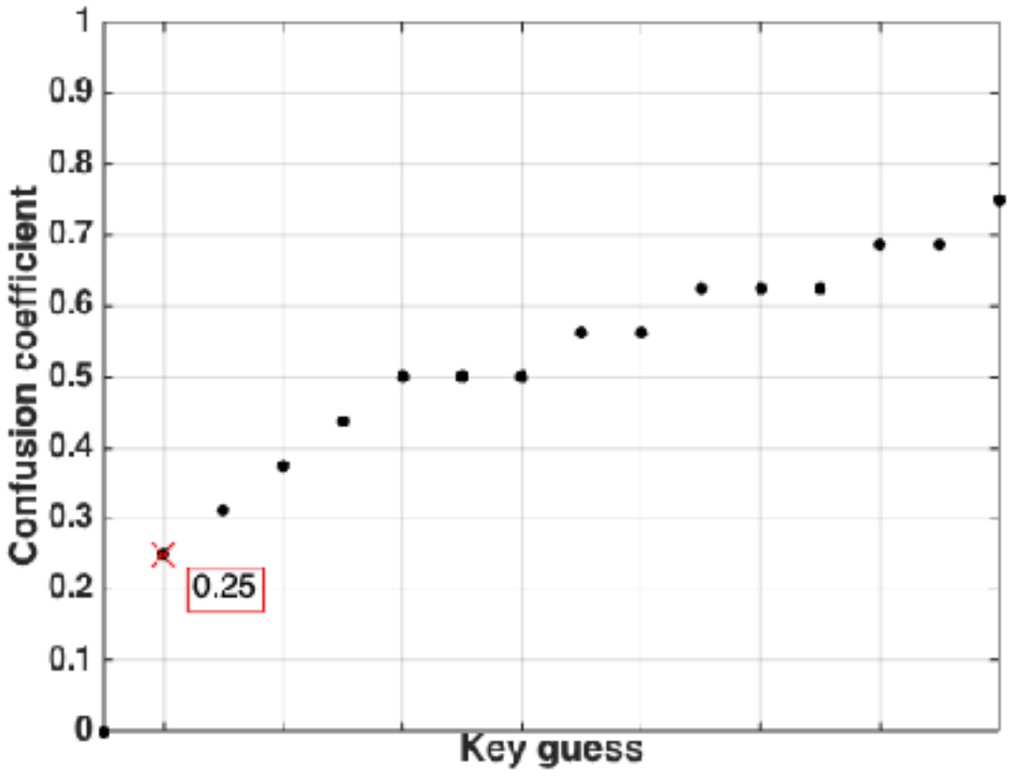
Piccolo



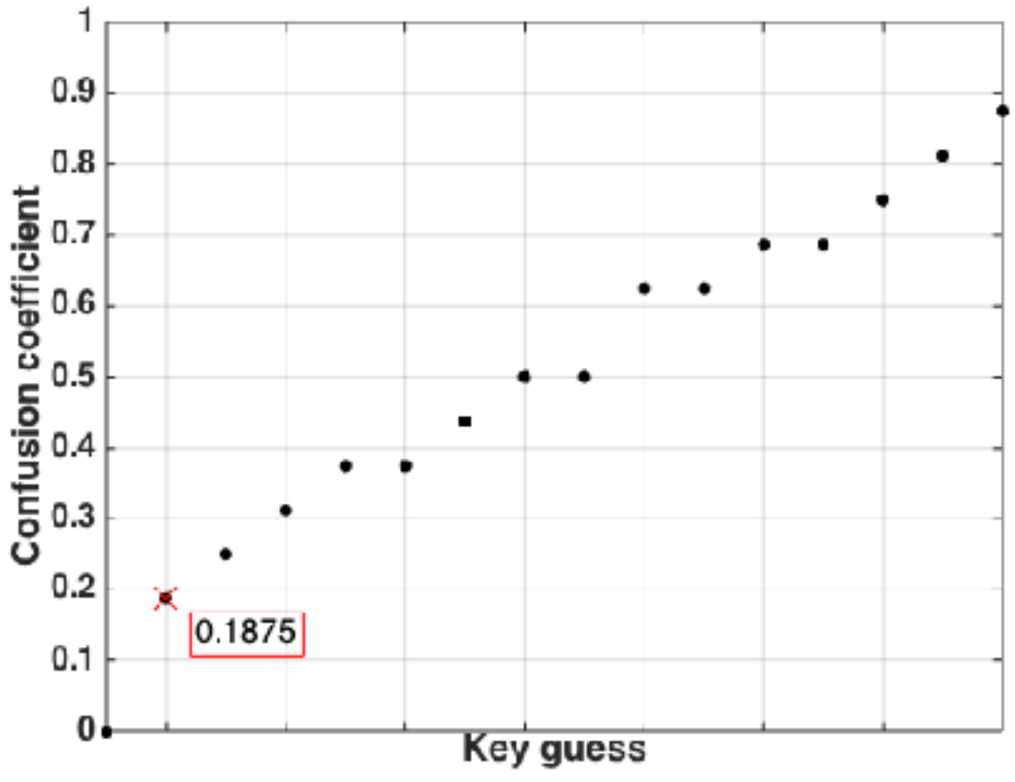
PRESENT / LED

Confusion Coefficients

the lower the minimum the higher the resistance



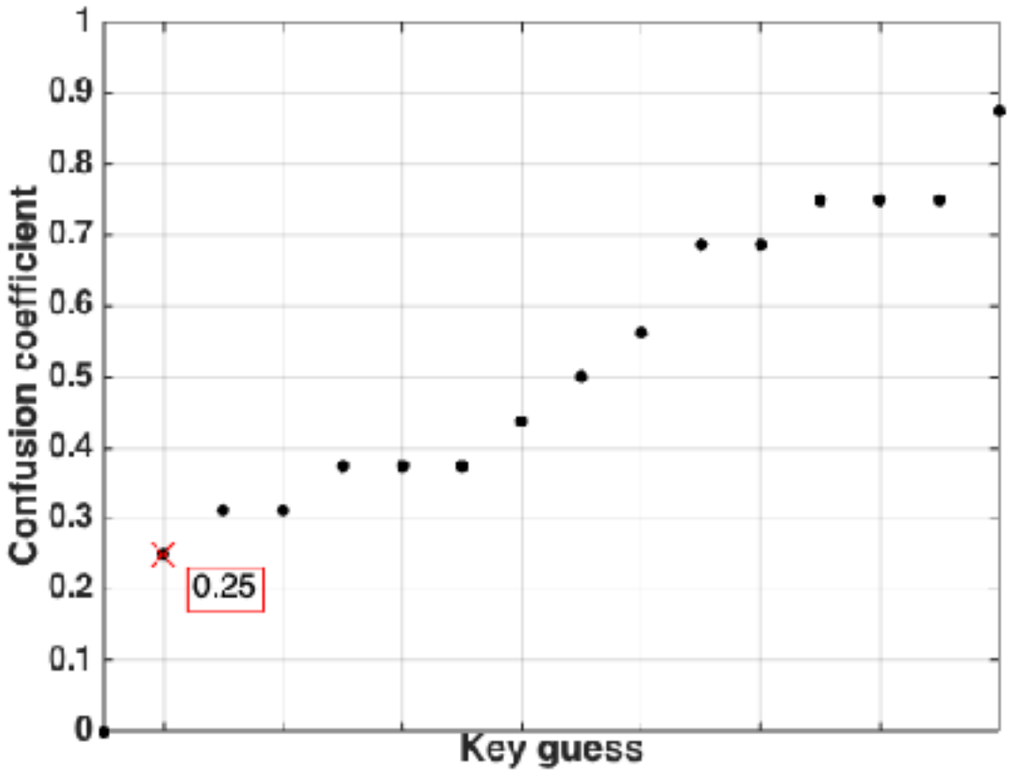
PRIDE



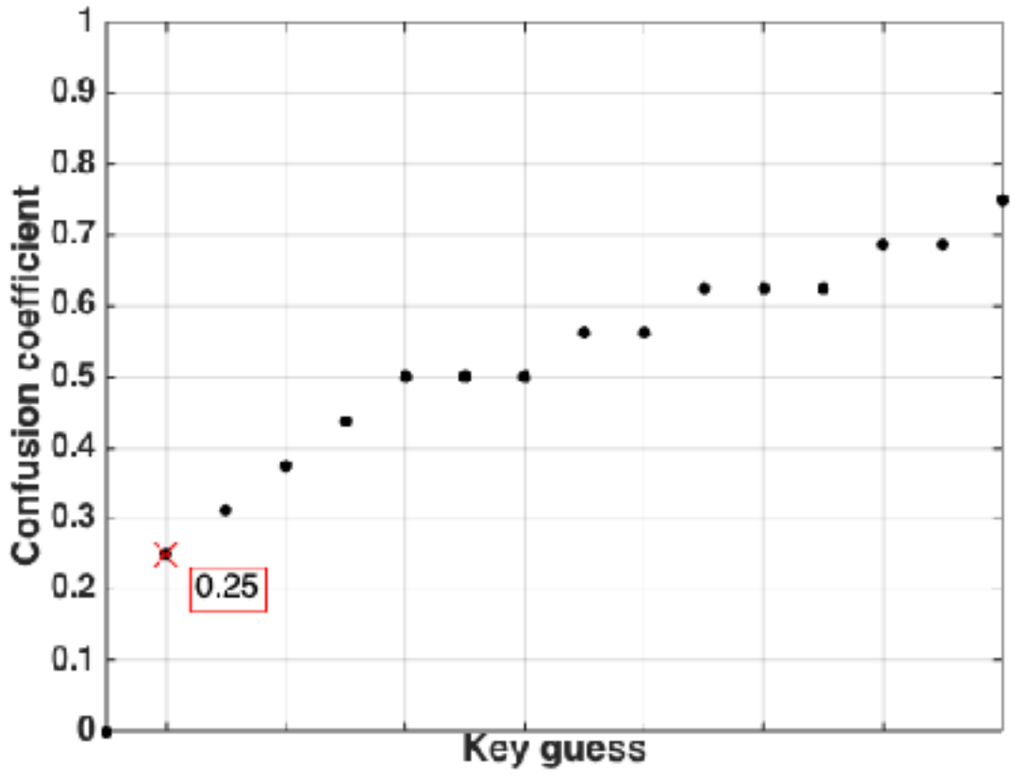
PRINCE

Confusion Coefficients

the lower the minimum the higher the resistance



RECTANGLE

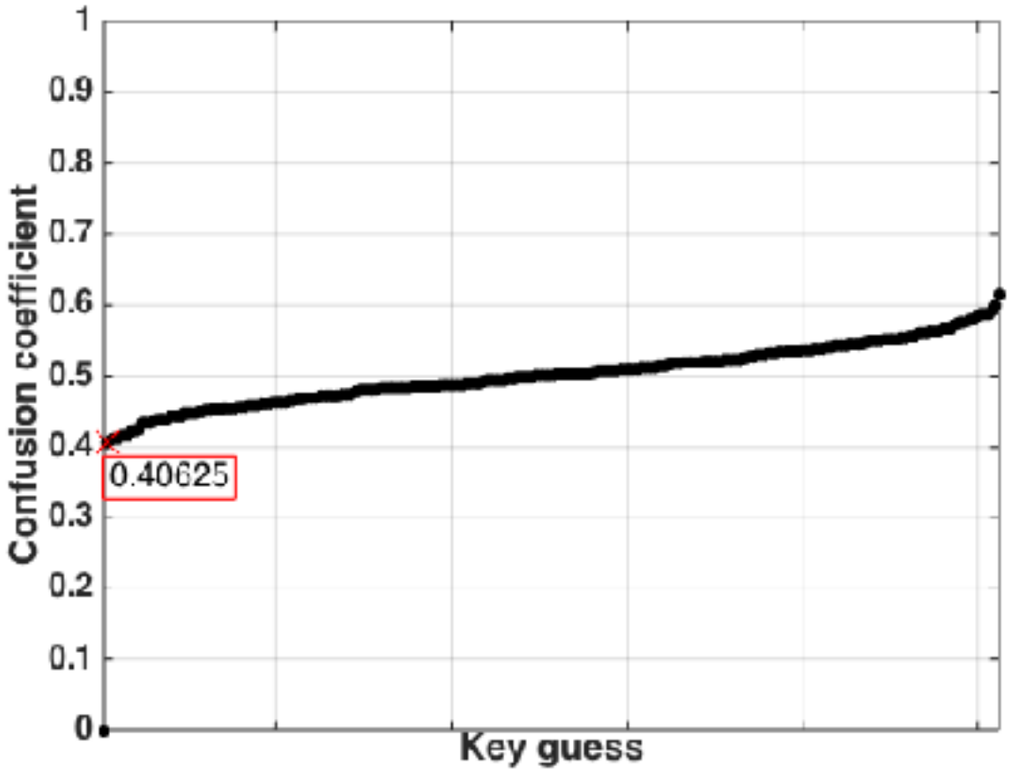


SKINNY

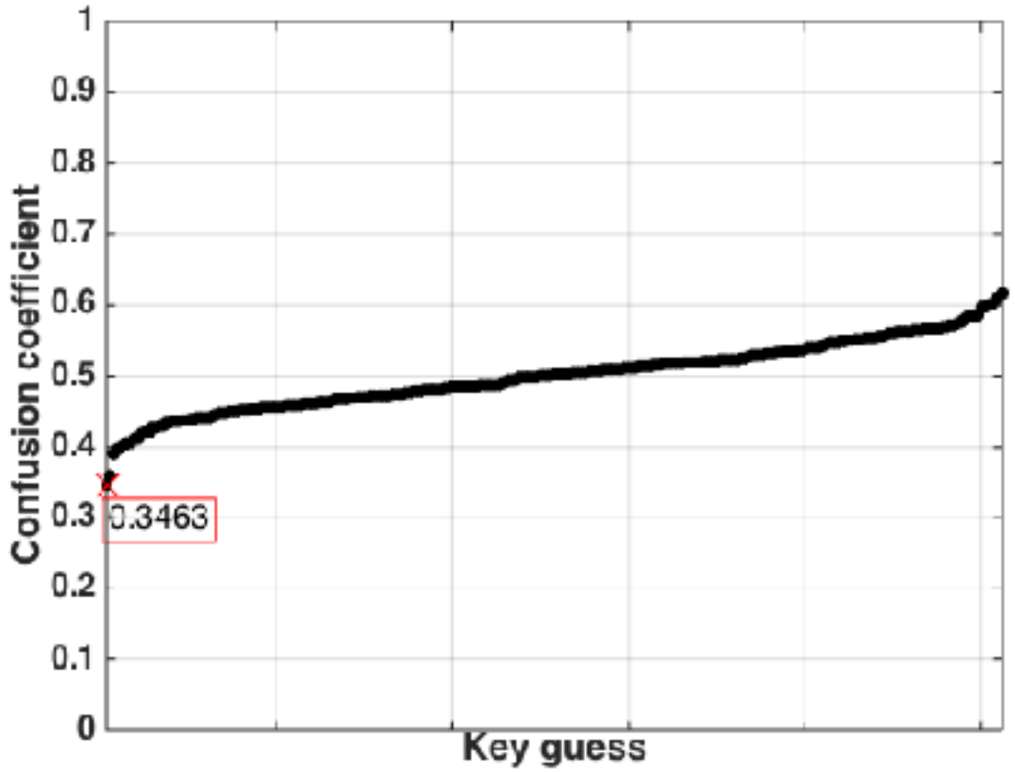
Confusion Coefficients

name	minimum
KLEIN	0.125
Midori 1	0.125
Midori 2	0.25
Mysterion	0.3125
Piccolo	0.375
PRESENT/LED	0.25
PRIDE	0.25
PRINCE	0.1875
RECTANGLE	0.25
SKINNY	0.25

Confusion Coefficients

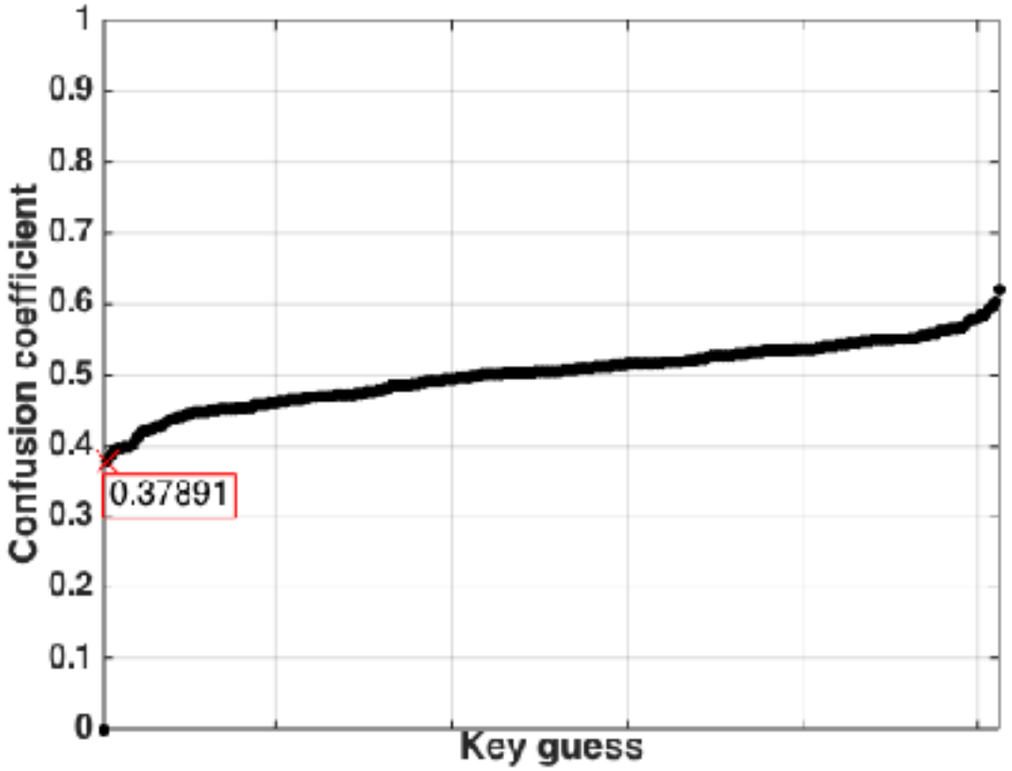


AES



Robin

Confusion Coefficients



Zorro

name	minimum
AES	0.4
Robin	0.34
Zorro	0.37

Confusion Coefficient

n = 4 (16 key hypotheses)

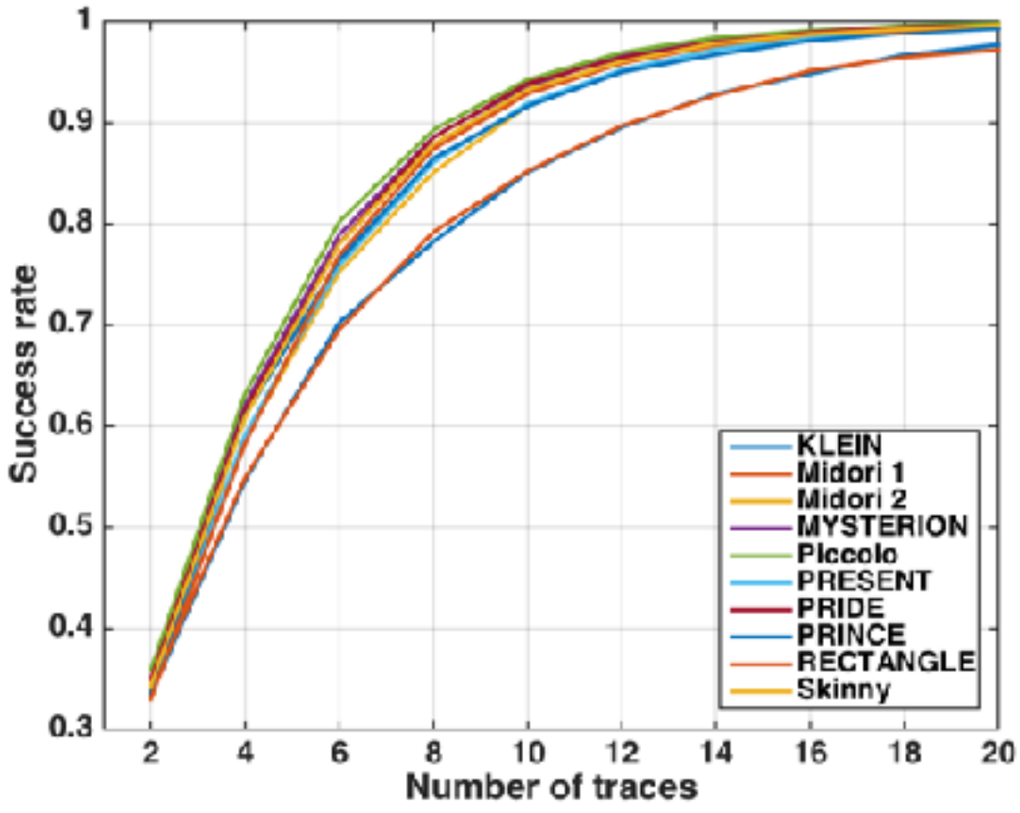
name	minimum
KLEIN	0.125
Midori 1	0.125
Midori 2	0.25
Mysterion	0.3125
Piccolo	0.375
PRESENT/LED	0.25
PRIDE	0.25
PRINCE	0.1875
RECTANGLE	0.25
SKINNY	0.25

n = 8 (256 key hypotheses)

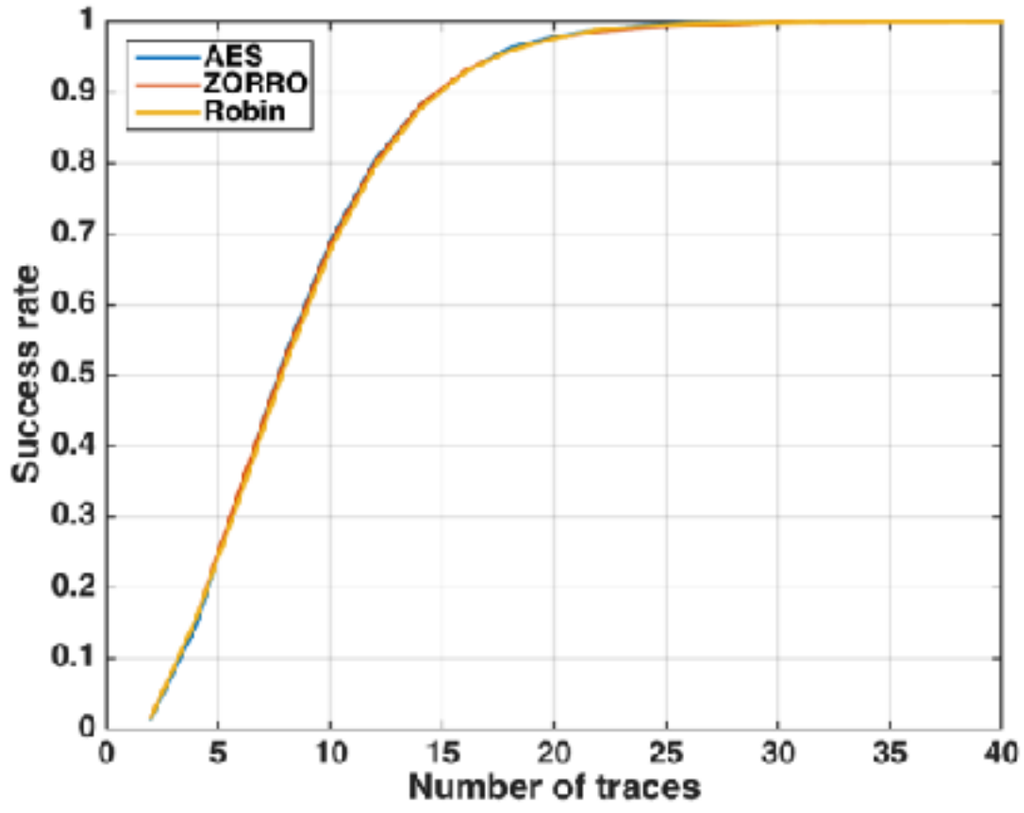
name	minimum
AES	0.4
Robin	0.34
Zorro	0.37

- SNR is different!!
- variance of the signal: n/4

Empirical Evaluation

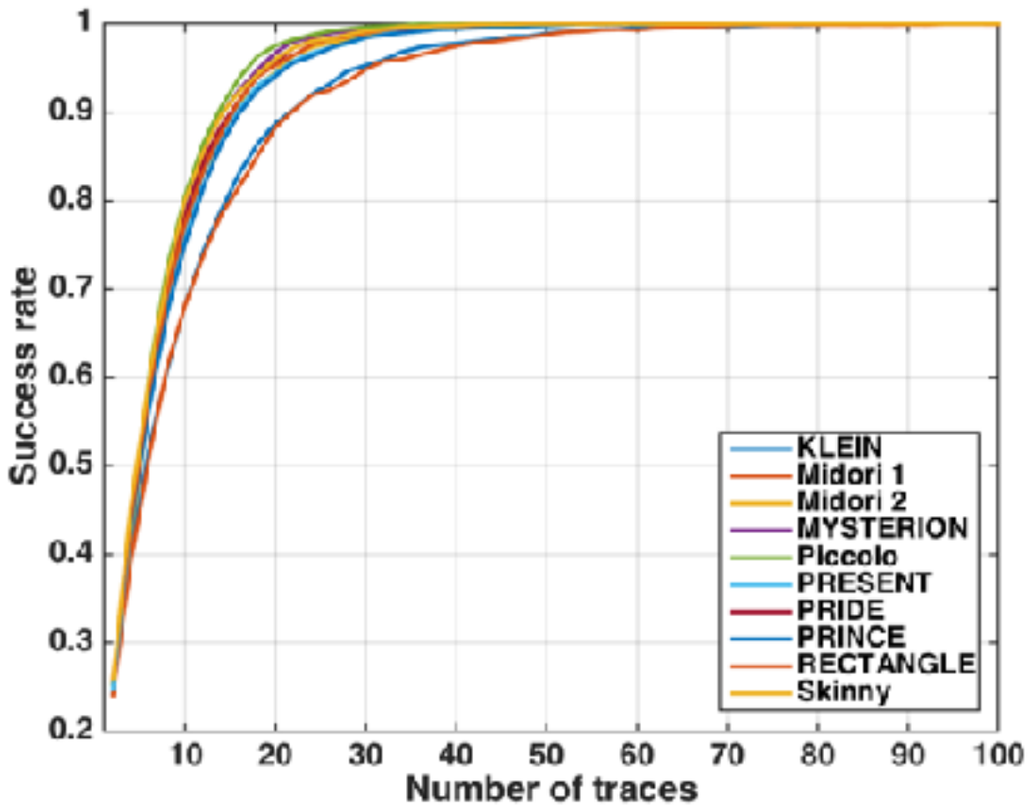


SNR = 2, sigma = sqrt(0.5)

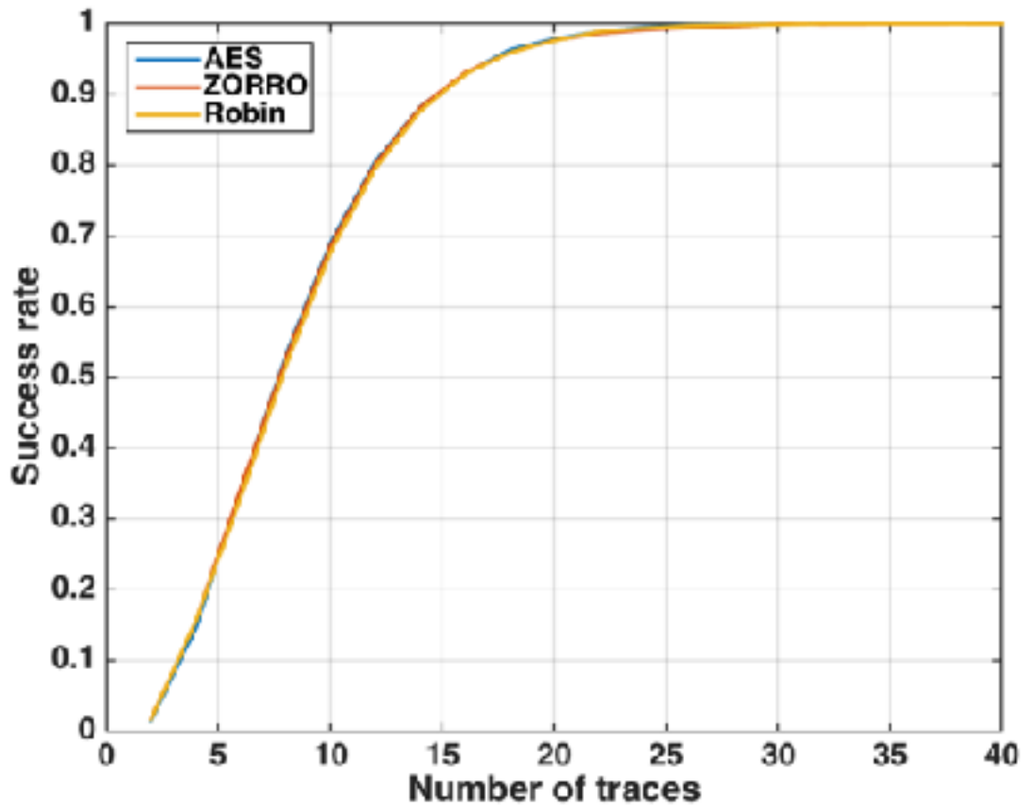


SNR = 2, sigma = 1

Empirical Evaluation

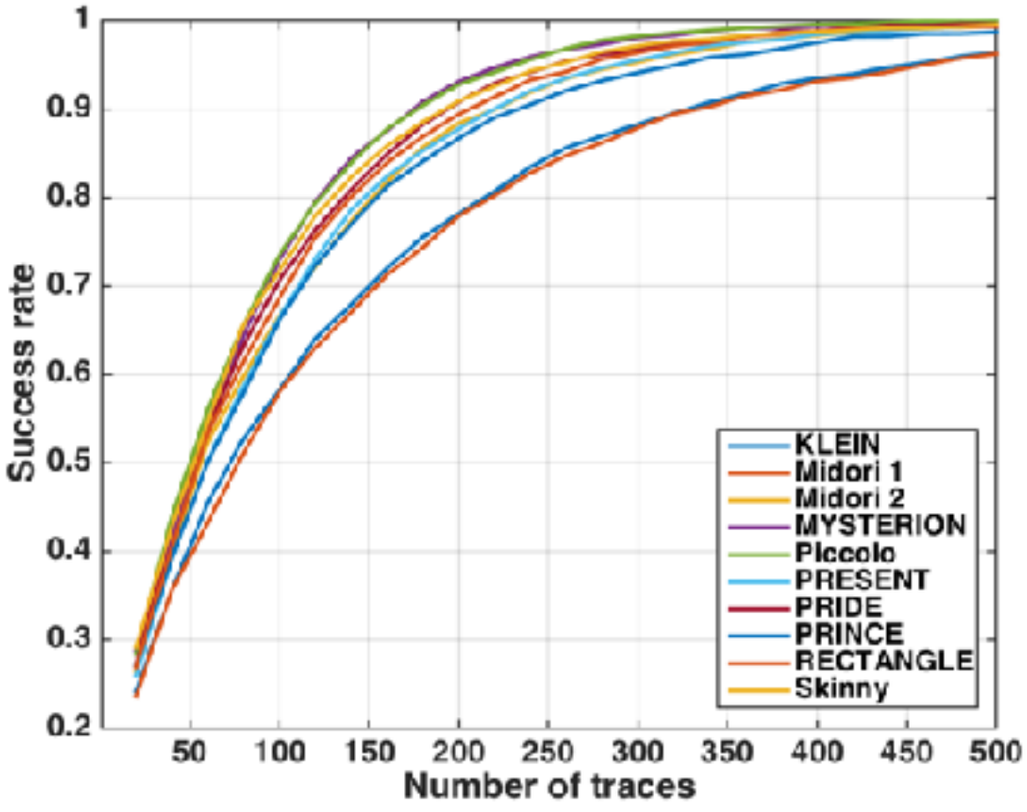


SNR = 1, **sigma = 1**

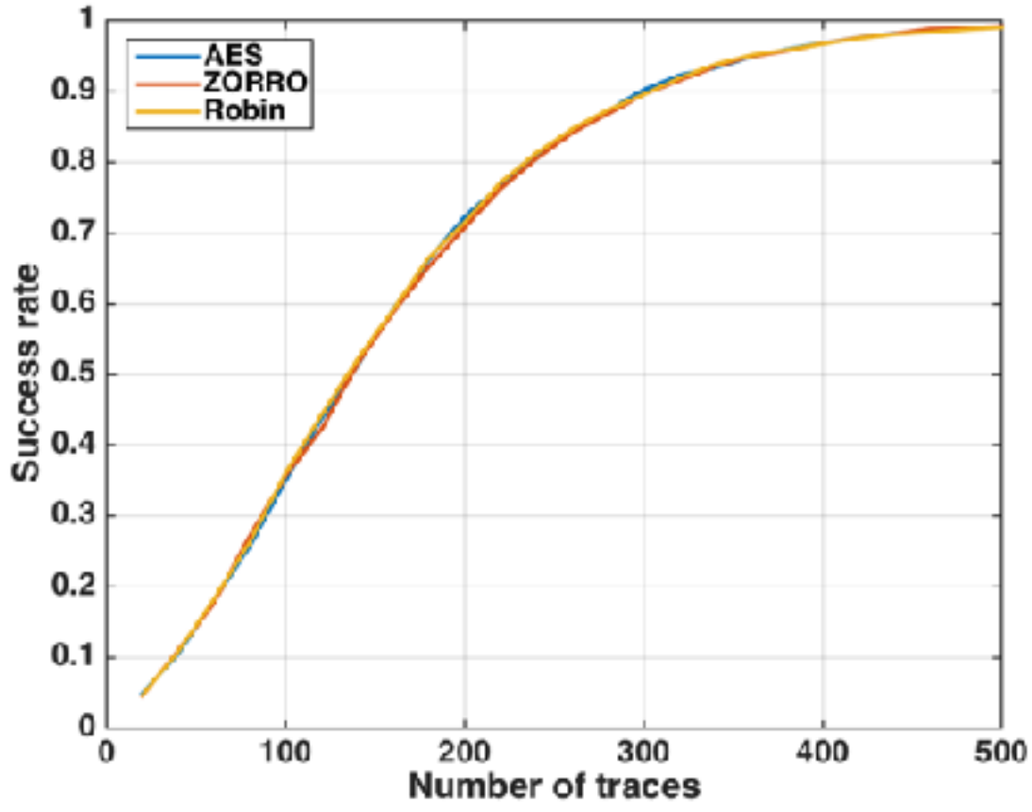


SNR = 2, **sigma = 1**

Empirical Evaluation

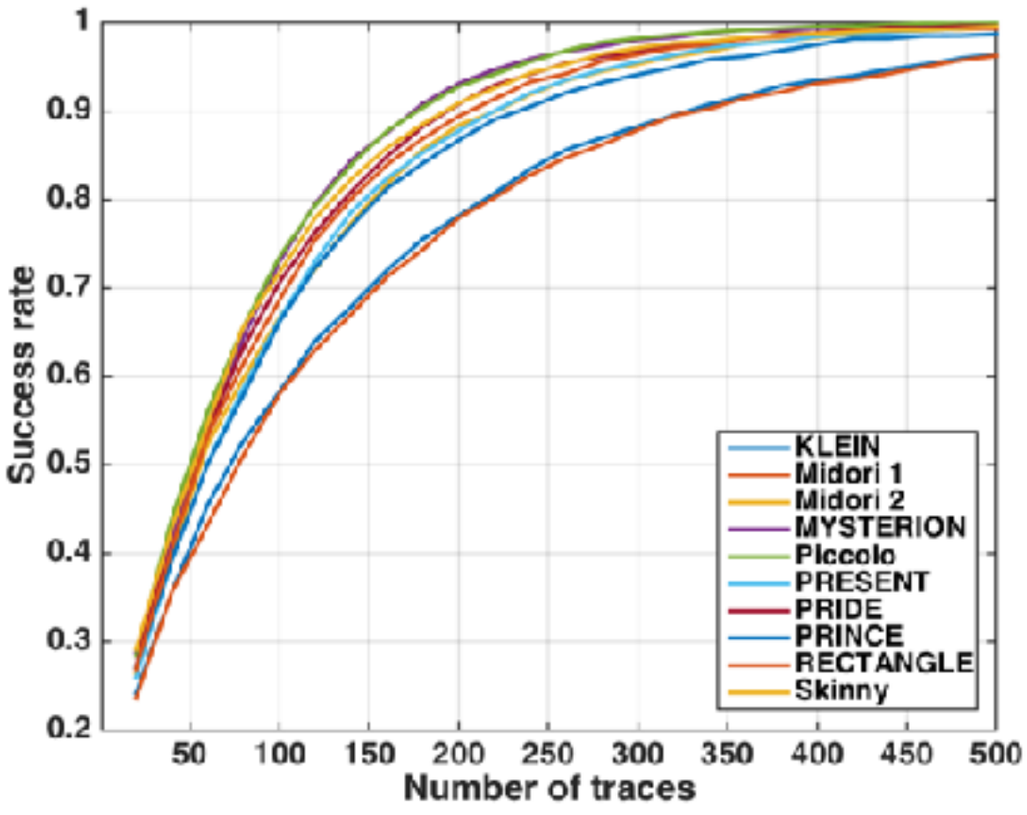


SNR = 1/16, sigma = 4

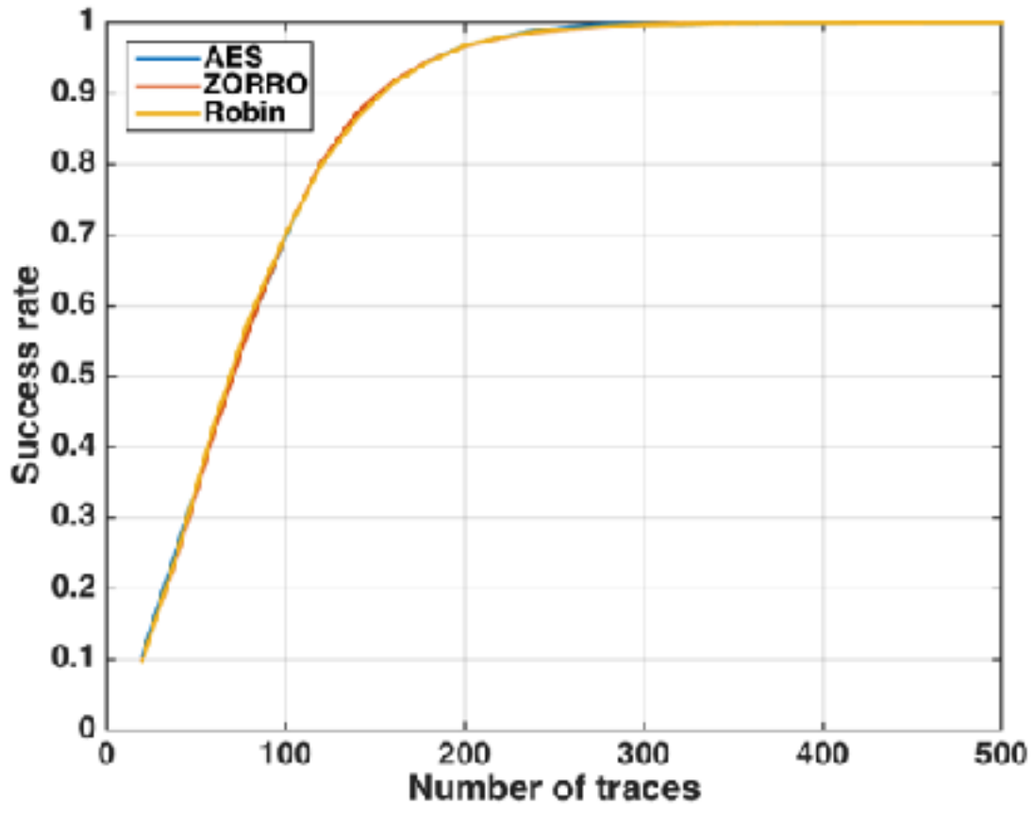


SNR =1/16, sigma = sqrt(32)

Empirical Evaluation



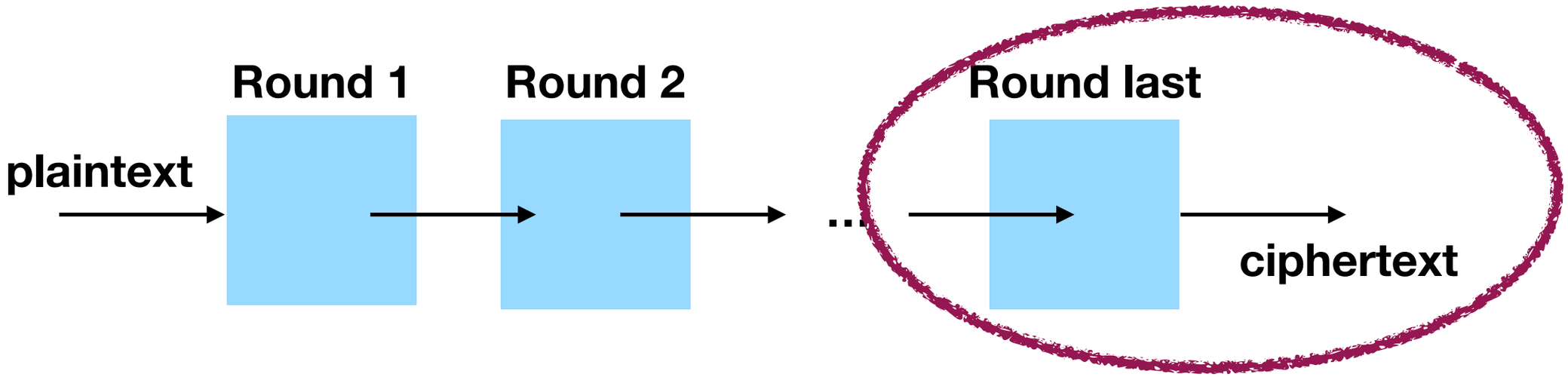
SNR = 1/16, **sigma = 4**



SNR =1/8, **sigma = 4**

Side-Channel Exploitation

- Success depends on the confusion coefficients of the inverse SBox

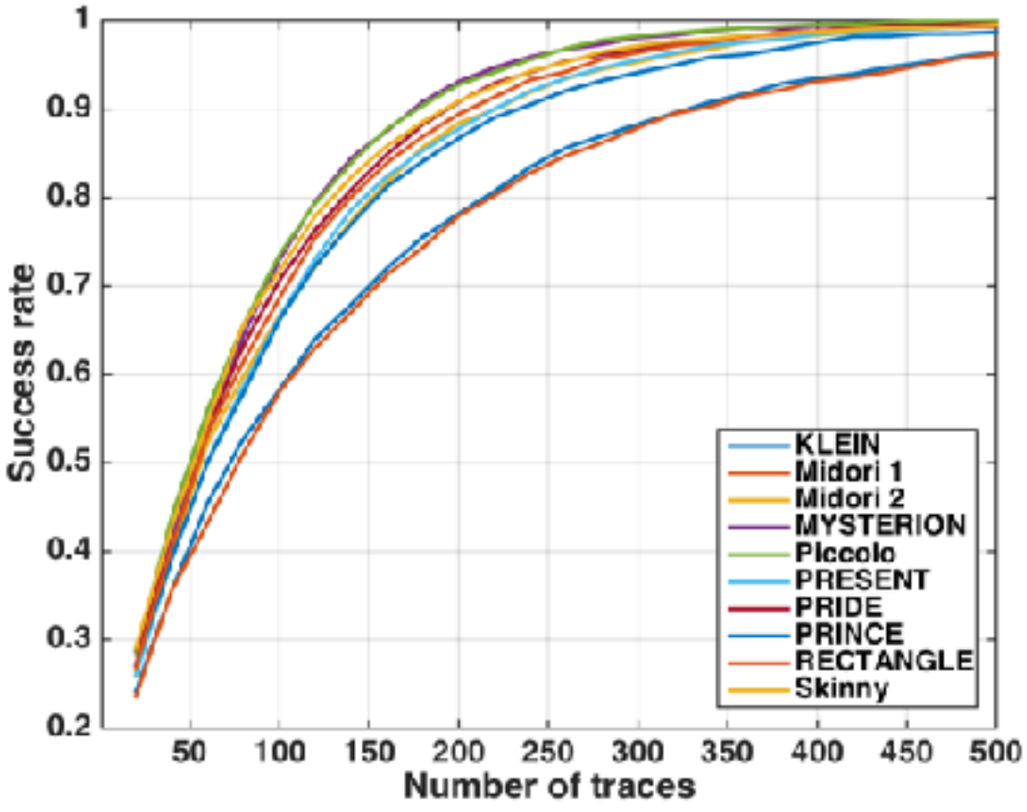


$$X = \mathrm{HW}(\mathtt{Sbox}^{-1}[C \oplus k^*]) + N$$

Confusion Coefficients

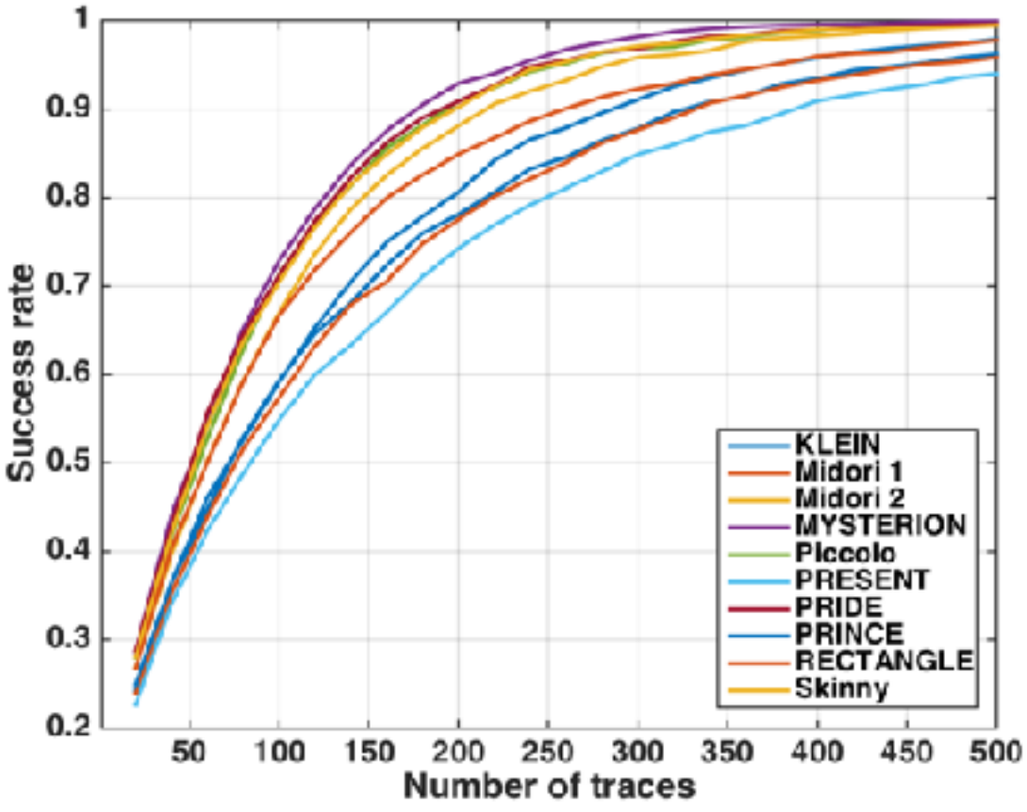
Name	involution	S-box	inverse S-box
KLEIN	x	0.125	0.125
Midori 1	x	0.125	0.125
Midori 2	x	0.250	0.250
Mysterion		0.3125	0.3125
Piccolo		0.375	0.25
PRESENT/LED		0.25	0.125
PRIDE	x	0.25	0.25
PRINCE		0.1875	0.1875
RECTANGLE		0.250	0.125
SKINNY	x	0.250	0.250
AES		0.406	0.388
Robin	x	0.347	0.347
Zorro		0.378	0.402

Empirical Evaluation



SNR = 1/16, sigma = 4

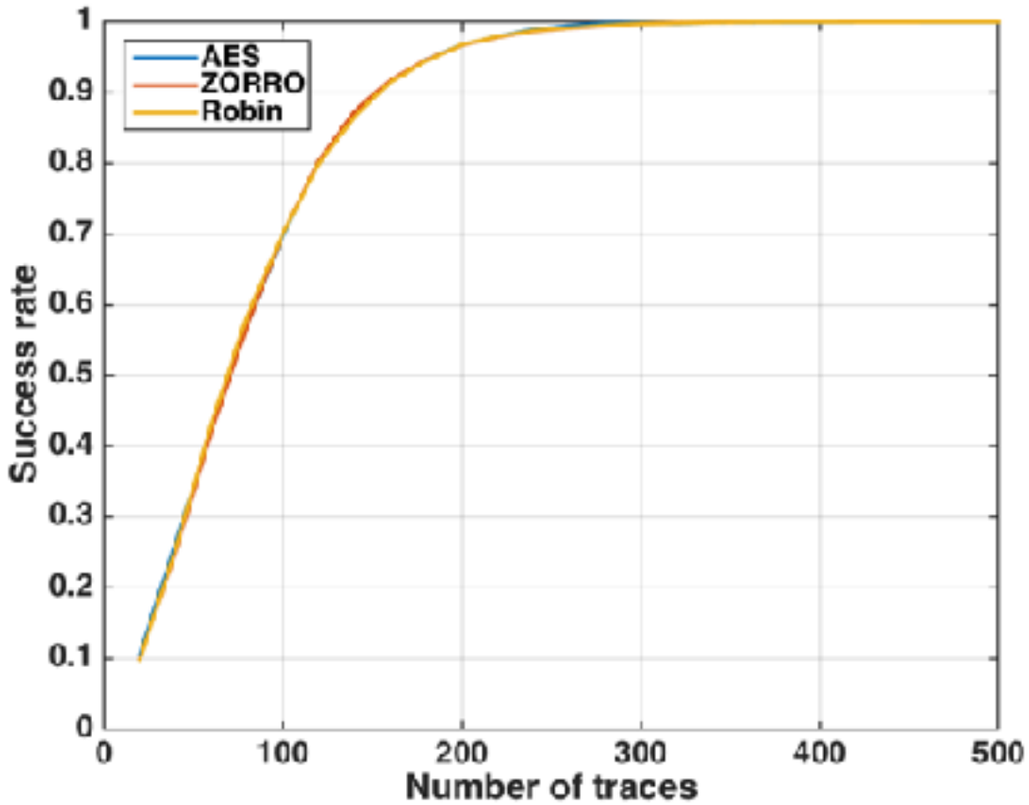
first round



SNR = 1/16, sigma = 4

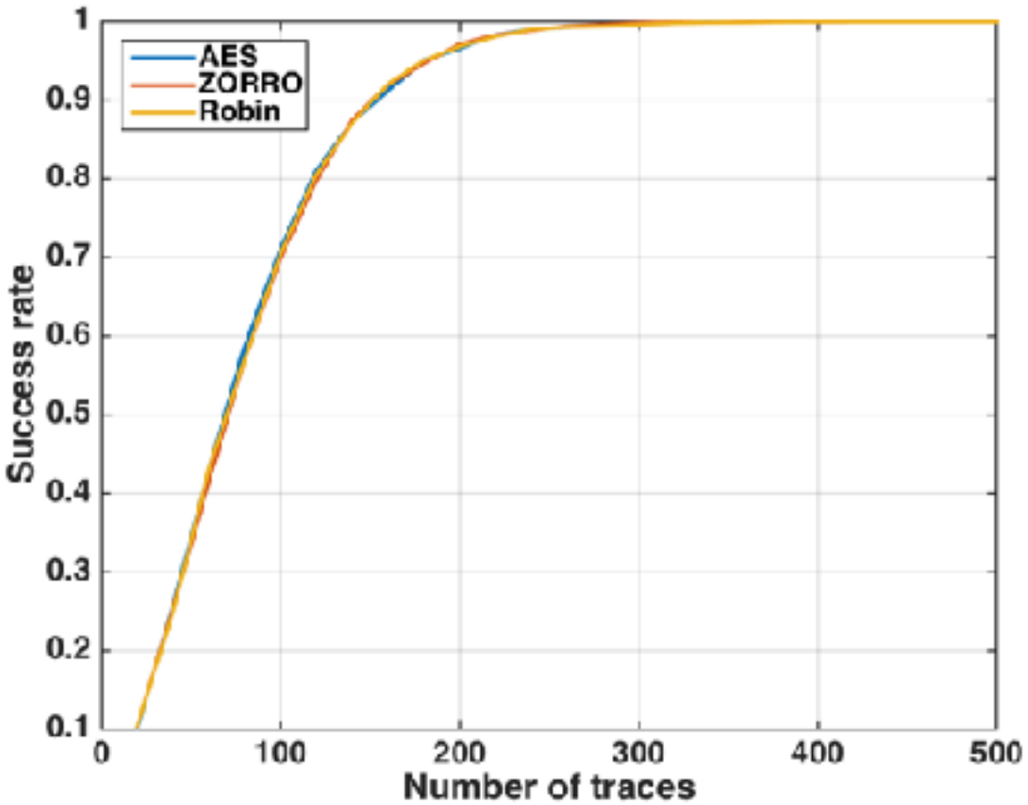
last round

Empirical Evaluation



SNR = 1/8, sigma = 4

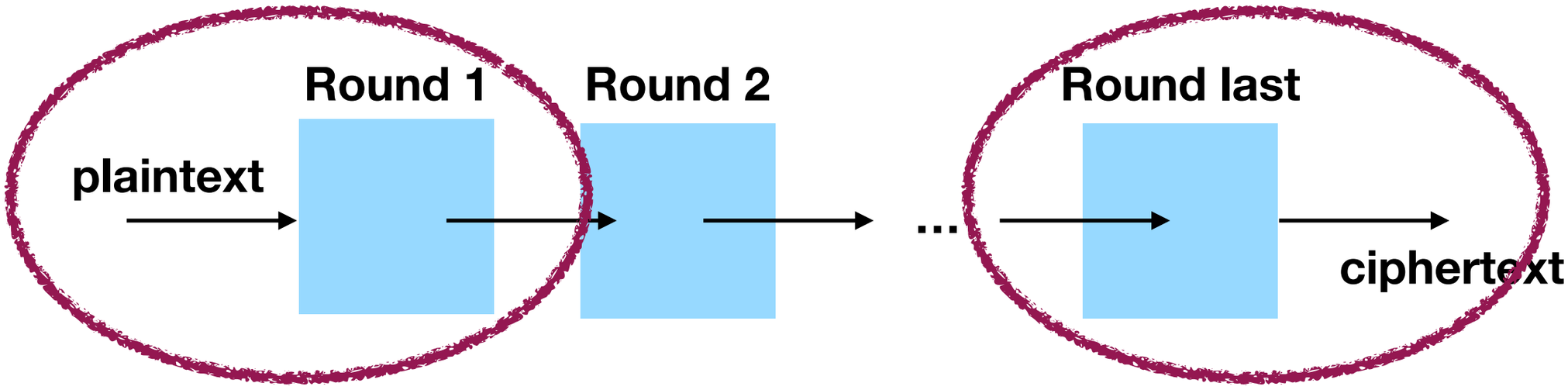
first round



SNR = 1/8, sigma = 4

last round

Side-Channel Exploitation

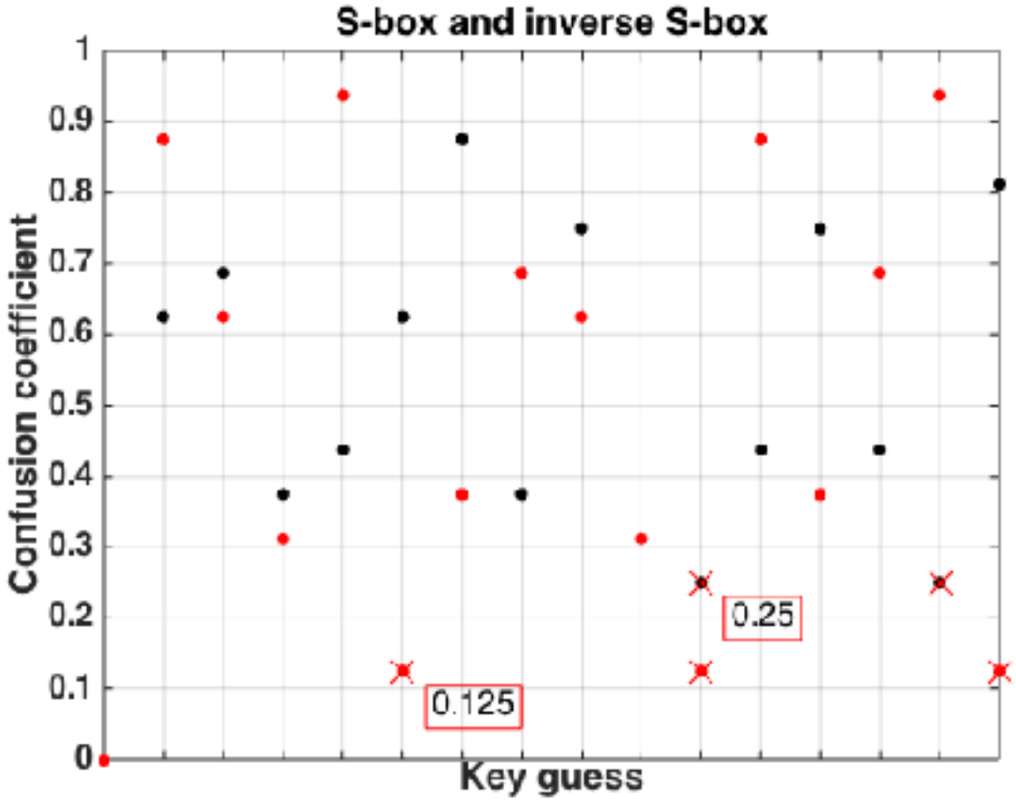


- Success depends on the combination of both (SBox and inverse SBox) confusion coefficients

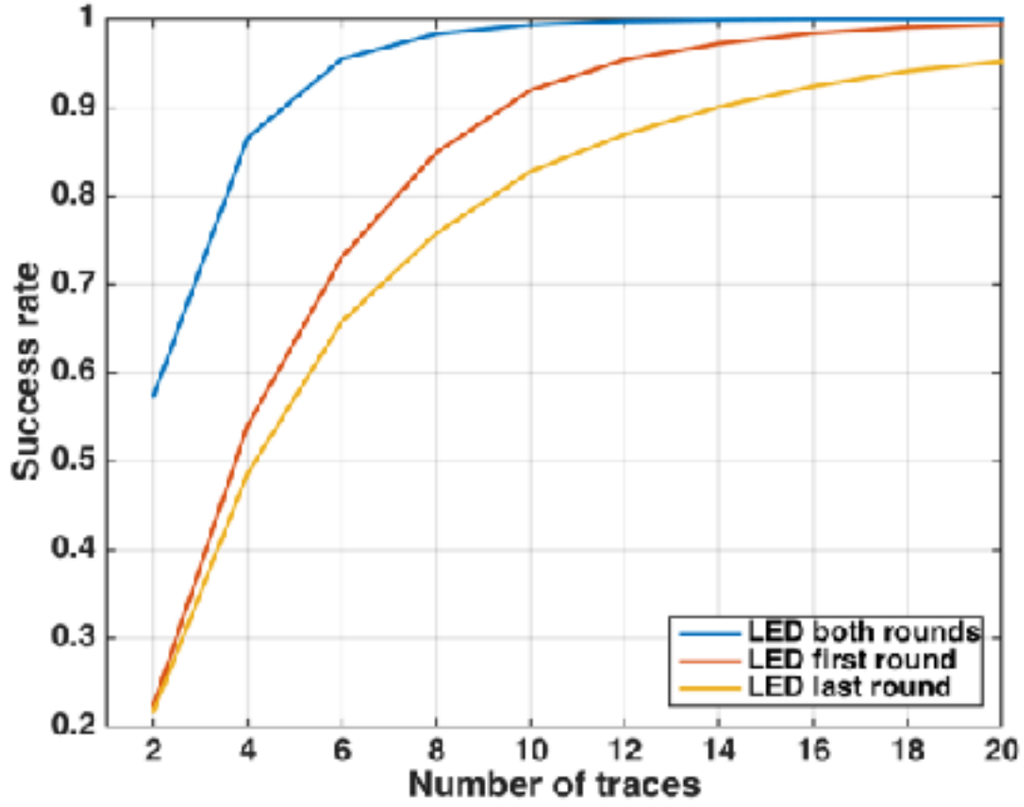
- (when round keys are straightforward computable from each other)

Side-Channel Exploitation

- Example with LED block cipher (lightweight key scheduling)



LED



SNR = sqrt(1/2), sigma = 2

Conclusion

- Basics of Power/EM side-channel leakage
- Where to attack AES and why
- Template attack / stochastic approach
- Confusion coefficient of 4-bit Sboxes
- Stayed tuned … next talk tomorrow:
  - more details on accuracy vs GE/SR
  - How to learn with imbalanced data
  - Redefinition of profiled attacks through semi-supervised learning
  - How to compare profiled attacks => Efficient Attacker Model

Introduction to (profiled) side-channel analysis

Annelie Heuser