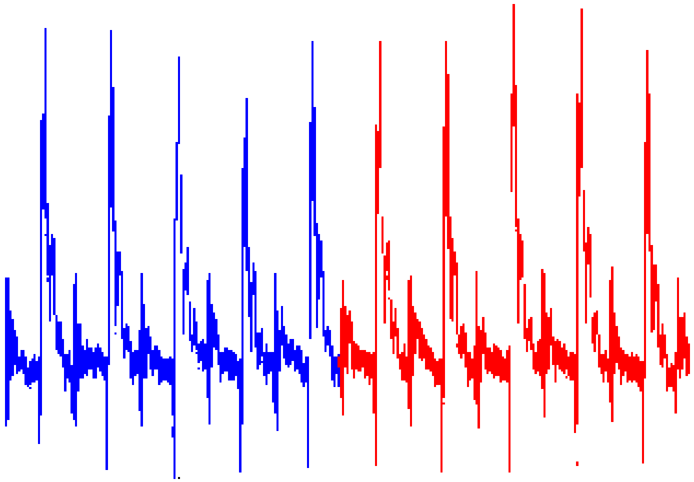
Towards Leakage Simulators that Withstand the Correlation Distinguisher



P. Pessl, ***F.-X. Standaert***, S. Mangard, F. Durvaux

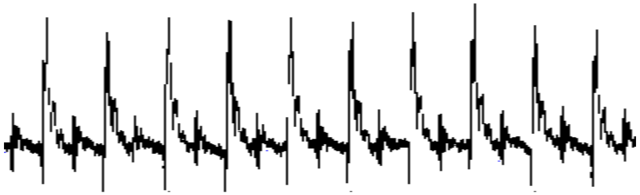
IAIK TU Graz (Austria), UCL Crypto Group (Belgium)

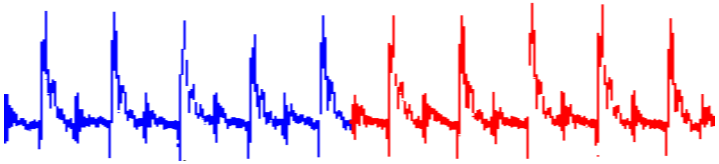
ASIACRYPT rump session, December 2014

Background

- Split & Concatenate Simulator (CRYPTO 2013)

$$\mathrm{L}(x, k, y) \approx \mathrm{L}(x, \tilde{k}, y^*) || \mathrm{L}(x^*, \tilde{k}, y)$$





Background

- Split & Concatenate Simulator (CRYPTO 2013)

$$\mathrm{L}(x, k, y) \approx \mathrm{L}(x, \tilde{k}, y^*) || \mathrm{L}(x^*, \tilde{k}, y)$$

- Longo Galea et al (ASIACRYPT 2014): ∃ correlation between samples *within* real traces (e.g. $\rho > 0.5$) … that are significantly reduced in simulated ones
  ⇒ Allows distinguishing!

Background

- Split & Concatenate Simulator (CRYPTO 2013)

$$\mathrm{L}(x, k, y) \approx \mathrm{L}(x, \tilde{k}, y^*) || \mathrm{L}(x^*, \tilde{k}, y)$$

- Longo Galea et al (ASIACRYPT 2014): ∃ correlation between samples *within* real traces (e.g. $\rho > 0.5$) … that are significantly reduced in simulated ones
  ⇒ Allows distinguishing!

- Proposed solution: very noisy implementations, *but it scales badly*: noise arbitrarily reduced with averaging

Background

- Split & Concatenate Simulator (CRYPTO 2013)

$$\mathrm{L}(x, k, y) \quad \approx \quad \mathrm{L}(x, \tilde{k}, y^*) || \mathrm{L}(x^*, \tilde{k}, y)$$

- Longo Galea et al (ASIACRYPT 2014): ∃ correlation between samples *within* real traces (e.g. $\rho > 0.5$)
  … that are significantly reduced in simulated ones
  ⇒ Allows distinguishing!

- Proposed solution: very noisy implementations, *but it scales badly*: noise arbitrarily reduced with averaging

Can we do better?

Origin of the intra-trace correlation

- Algorithmic? Unlikely: $\rho\big(x, \mathrm{Sbox}(x)\big) \ll 0.5$

Origin of the intra-trace correlation

- Algorithmic? Unlikely: $\rho\left(x, \mathrm{Sbox}(x)\right) \ll 0.5$
- Physical then $\Rightarrow$ let’s use a simple physical model

Origin of the intra-trace correlation

- Algorithmic? Unlikely: $\rho\big(x, \mathrm{Sbox}(x)\big) \ll 0.5$
- Physical then ⇒ let's use a simple physical model

$$\mathrm{L}(x,k,y) = \underbrace{\delta(x,k,y)}_{\text{signal}} + \underbrace{N}_{\text{noise}}$$

Origin of the intra-trace correlation

- Algorithmic? Unlikely: $\rho\big(x, \mathrm{Sbox}(x)\big) \ll 0.5$
- Physical then ⇒ let's use a simple physical model

$$\mathrm{L}(x,k,y) = \underbrace{\delta(x,k,y)}_{\text{signal}} + \underbrace{N}_{\text{noise}}$$

⇒ Does the correlation come from signal or noise?

Origin of the intra-trace correlation

- Algorithmic? Unlikely: $\rho\big(x, \mathrm{Sbox}(x)\big) \ll 0.5$
- Physical then ⇒ let's use a simple physical model

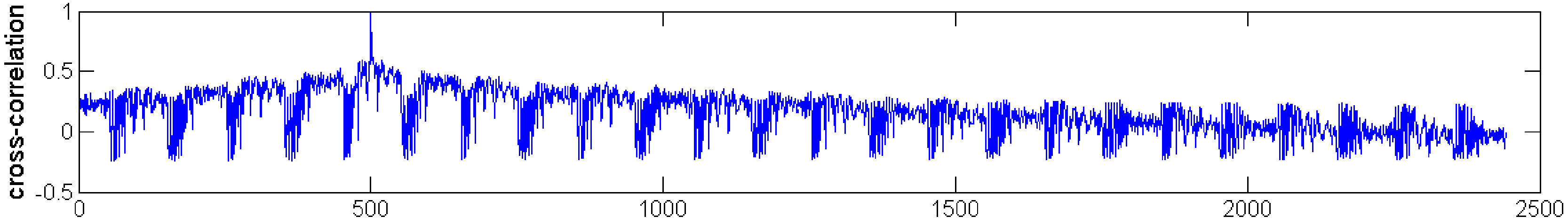
$$\mathrm{L}(x,k,y) = \underbrace{\delta(x,k,y)}_{\text{signal}} + \underbrace{N}_{\text{noise}}$$

⇒ Does the correlation come from signal or noise?

- In particular for *large parallel implementations* (since we know 8-bit AES implementations can be broken in one trace anyway – see SASCA paper)

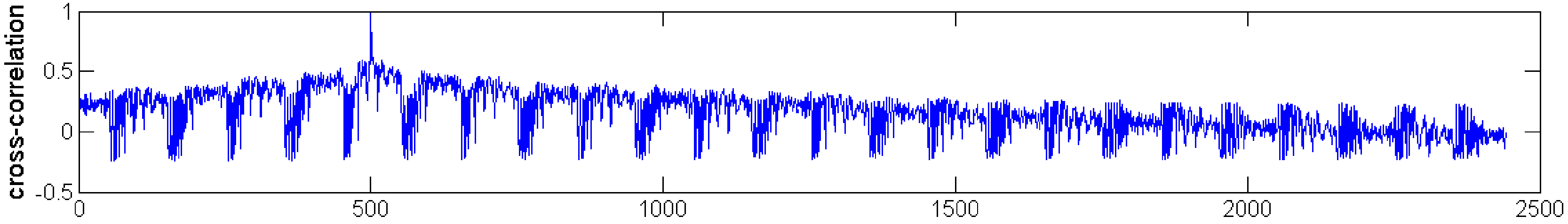
Repeating experiments with a 65nm ASIC

- Intra-trace correlation (real traces, sample 500)

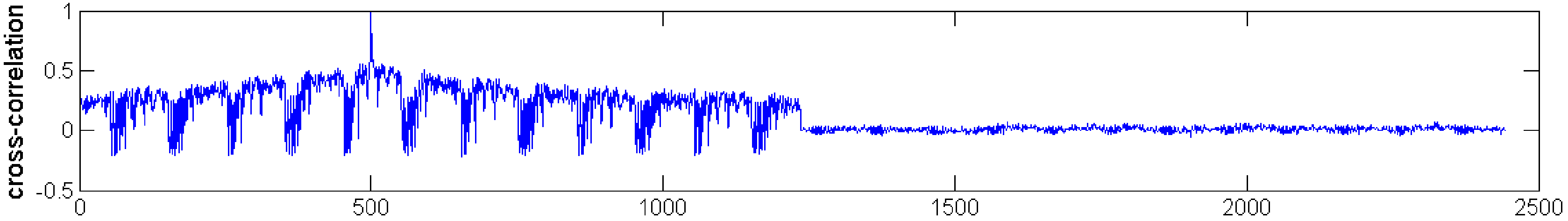


Repeating experiments with a 65nm ASIC

- Intra-trace correlation (real traces, sample 500)

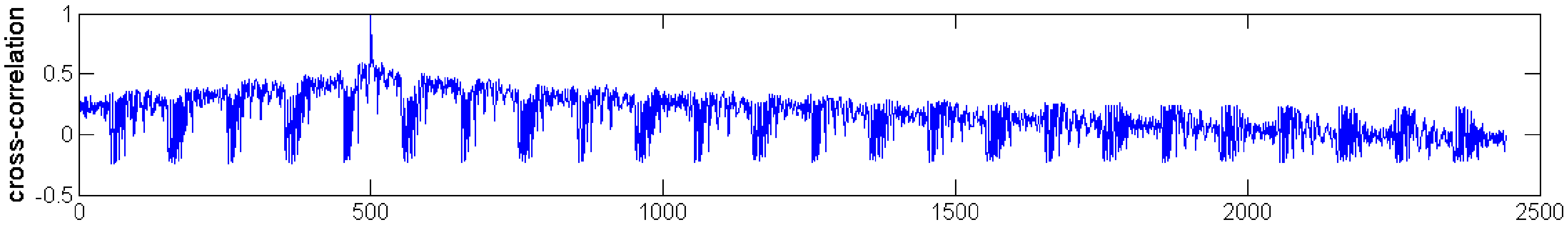


- Same, with simulated traces $\mathrm{L}(x, \tilde{k}, y^*) || \mathrm{L}(x^*, \tilde{k}, y)$

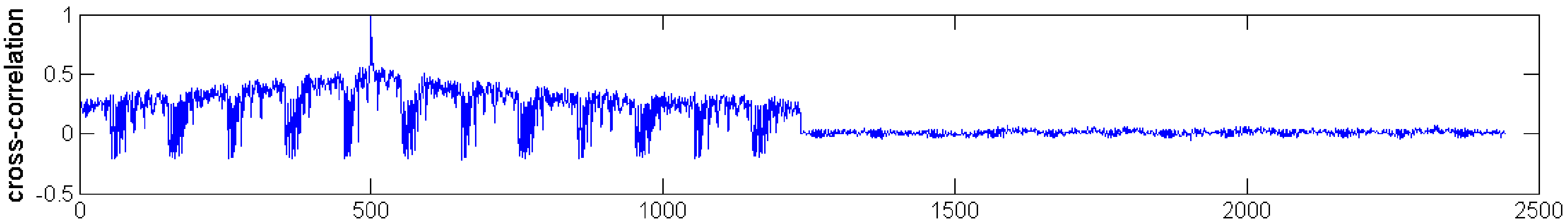


Repeating experiments with a 65nm ASIC

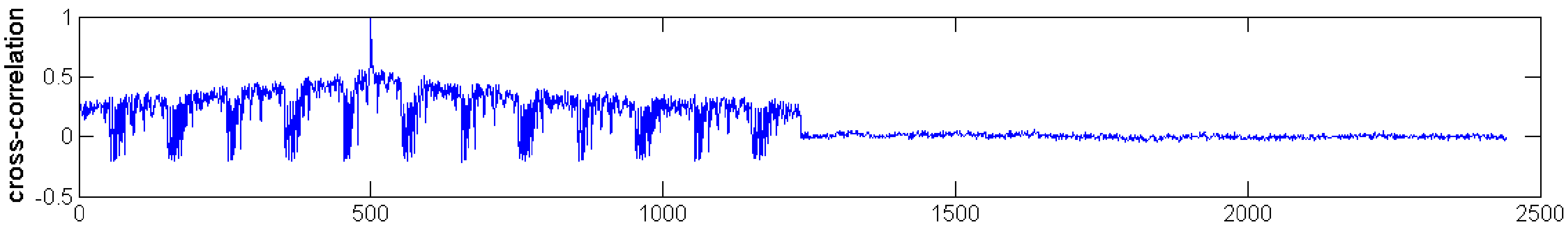
- Intra-trace correlation (real traces, sample 500)



- Same, with simulated traces $\mathrm{L}(x, \tilde{k}, y^*) || \mathrm{L}(x^*, \tilde{k}, y)$

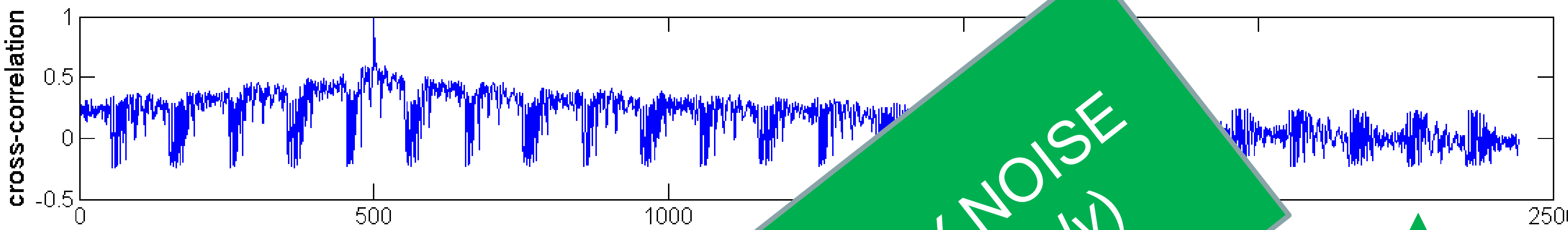


- & fake simulated traces $\delta(x, k, y) + N_1 || \delta(x, k, y) + N_2$



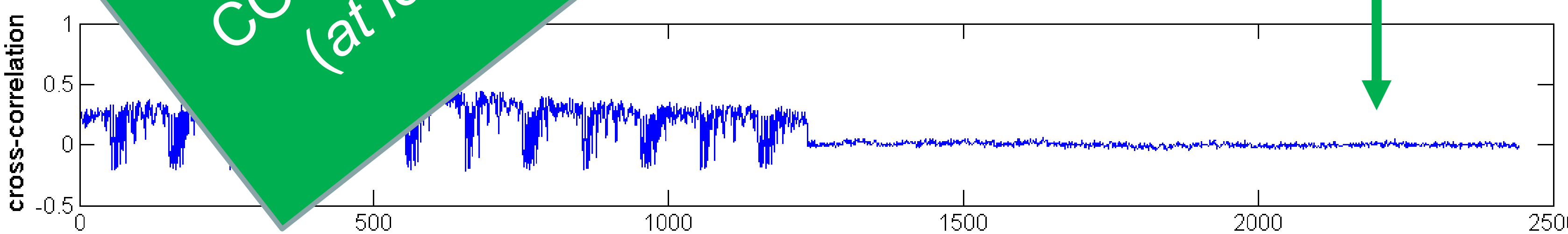
Repeating experiments with a 65nm ASIC

- Intra-trace correlation (real traces, sample 500)



- Same, with simulat… $y^*)||\mathrm{L}(x^*,\tilde{k},y)$

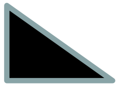
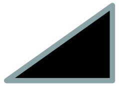
- & fak… es $\delta(x,k,y)+N_1||\delta(x,k,y)+N_2$



CORRELATION DOMINATED BY NOISE
(at least in this hardware case study)

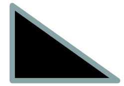
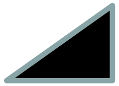
A first improvement

- Sliding simulator

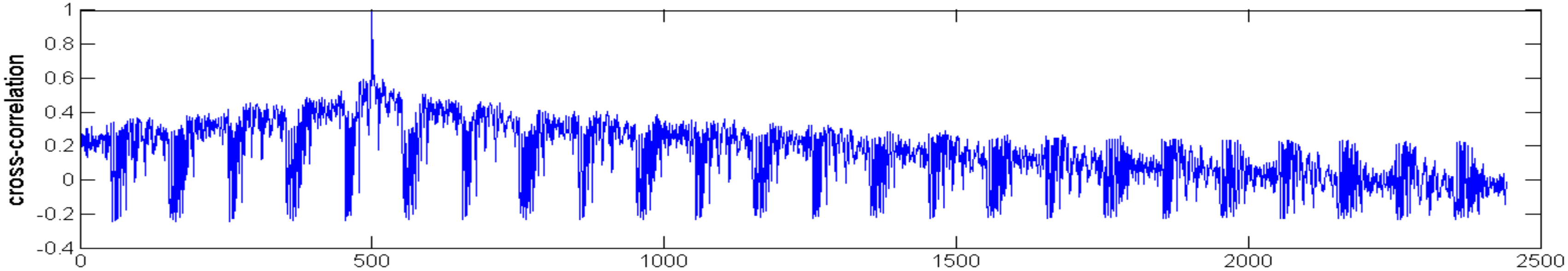
$L(x, \tilde{k}, y^*) \cdot$  $+ \ L(x^*, \tilde{k}, y) \cdot$ 

A first improvement

- Sliding simulator

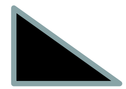
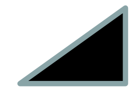
$L(x, \tilde{k}, y^*) \cdot$  $+ \; L(x^*, \tilde{k}, y) \cdot$ 

- Real traces

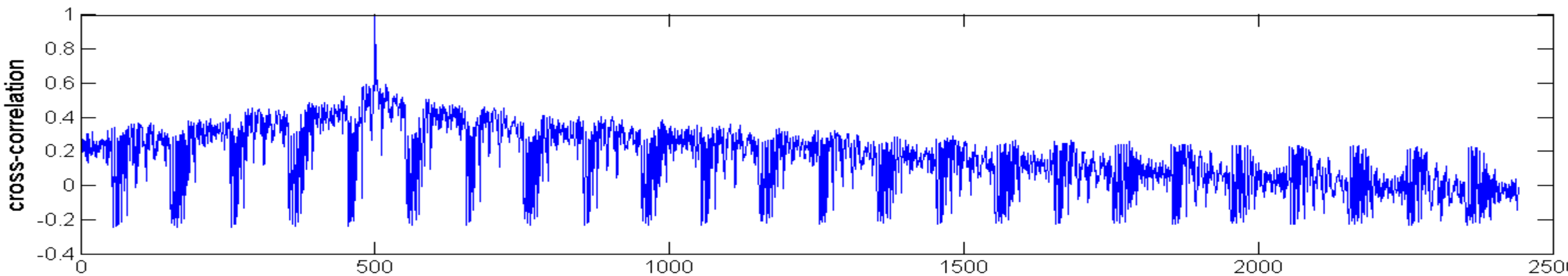


A first improvement

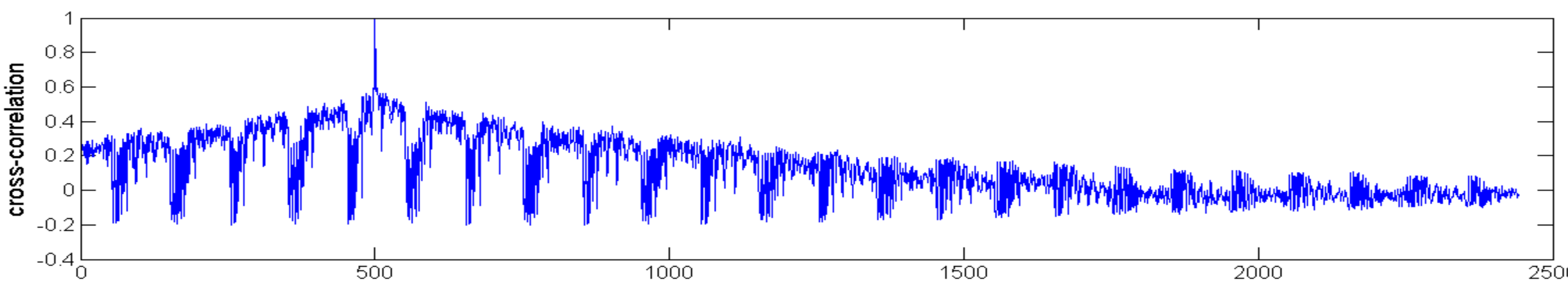
- Sliding simulator

$$L(x, \tilde{k}, y^*) \cdot$$  $$+ \; L(x^*, \tilde{k}, y) \cdot$$ 

- Real traces



- Simulated traces

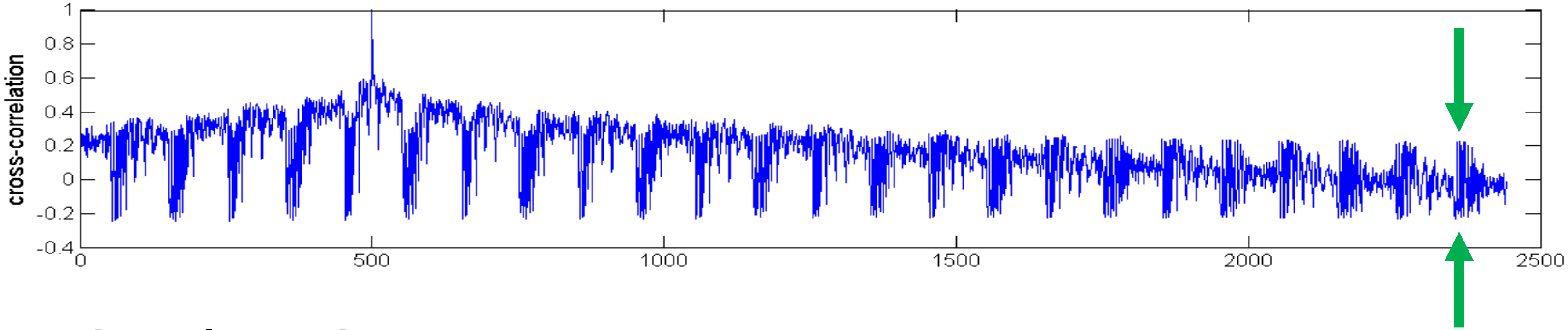


A first improvement

- Sliding simulator

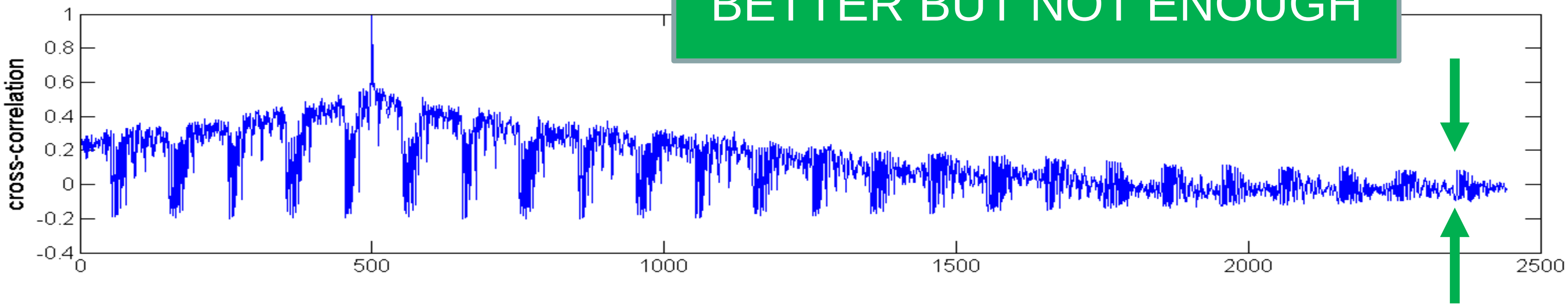
$$L(x, \tilde{k}, y^*) \cdot \qquad + \quad L(x^*, \tilde{k}, y) \cdot$$

- Real traces



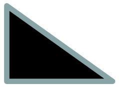
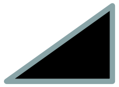
- Simulated traces

BETTER BUT NOT ENOUGH



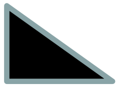
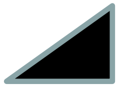
The main idea: separate signal and noise

- Sliding signal + noise simulator

$\hat{\delta}(x, \tilde{k}, y^*) \cdot$  $+ \ \hat{\delta}(x^*, \tilde{k}, y) \cdot$  $+ N$

The main idea: separate signal and noise

- Sliding signal + noise simulator

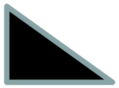
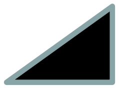
$\hat{\delta}(x, \tilde{k}, y^*) \cdot$  $+\ \hat{\delta}(x^*, \tilde{k}, y) \cdot$  $+\ N$

avg. trace

avg. trace – single trace

The main idea: separate signal and noise

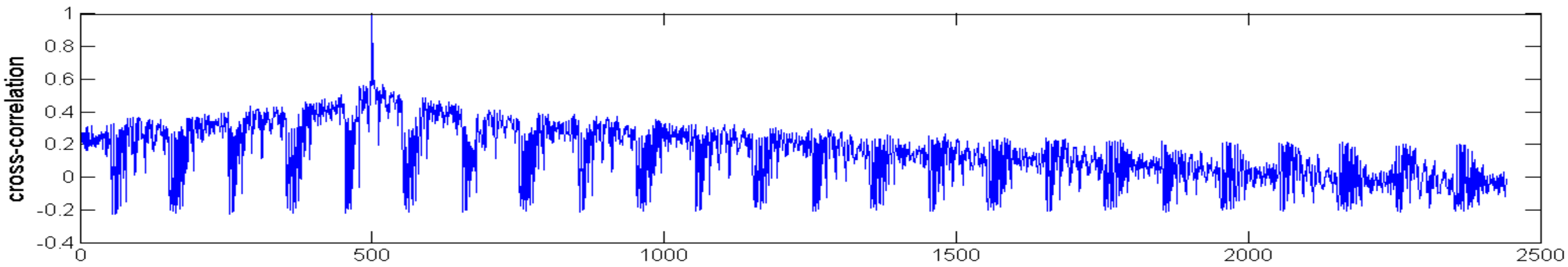
- Sliding signal + noise simulator

$\hat{\delta}(x, \tilde{k}, y^*) \cdot$  $+$ $\hat{\delta}(x^*, \tilde{k}, y) \cdot$  $+ N$

avg. trace

avg. trace – single trace

- Real traces



The main idea: separate signal and noise

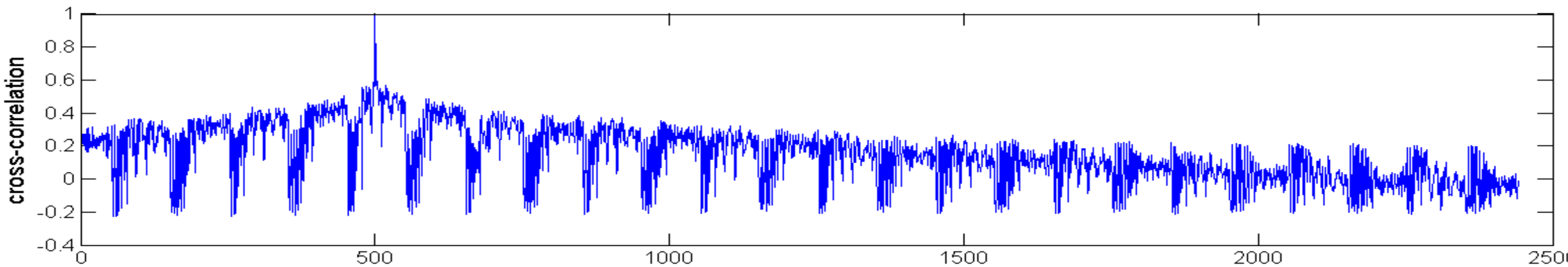
- Sliding signal + noise simulator

$$\hat{\delta}(x,\tilde{k},y^*)\cdot \quad + \quad \hat{\delta}(x^*,\tilde{k},y)\cdot \quad + N$$

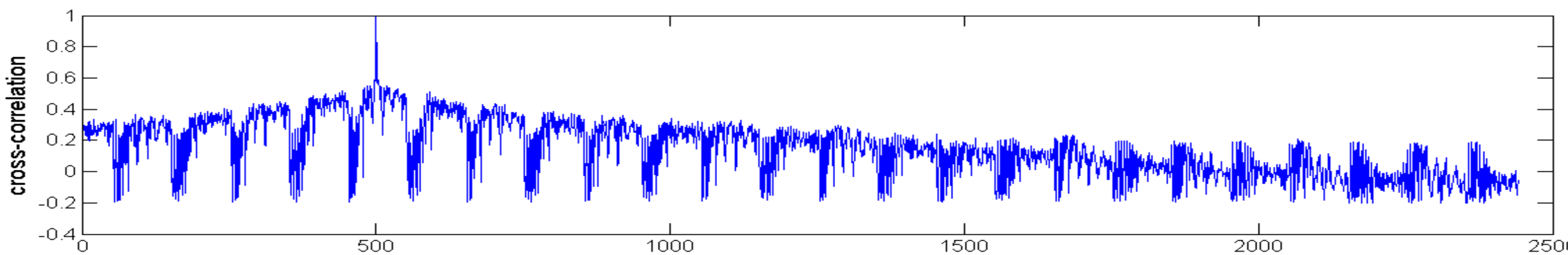
avg. trace

avg. trace – single trace

- Real traces

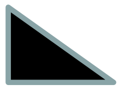
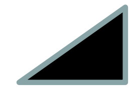


- Simulated traces



The main idea: separate signal and noise

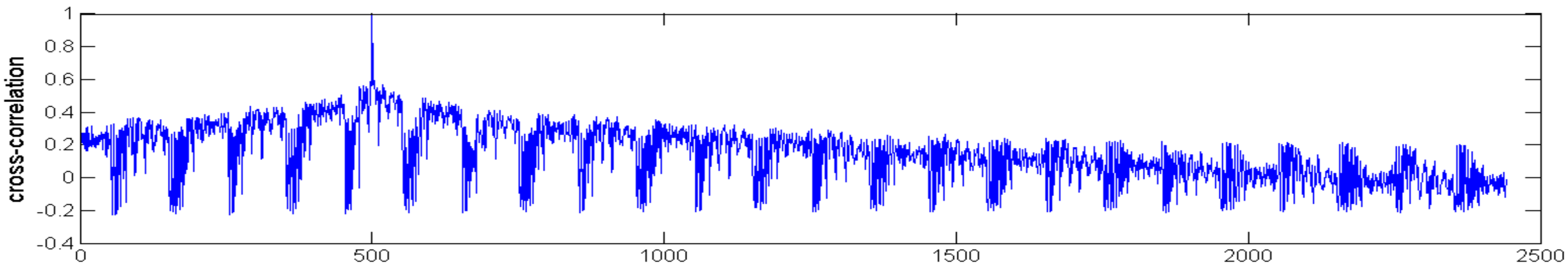
- Sliding signal + noise simulator

$\hat{\delta}(x, \tilde{k}, y^*) \cdot$  $+ \; \hat{\delta}(x^*, \tilde{k}, y) \cdot$  $+ \; N$

avg. trace

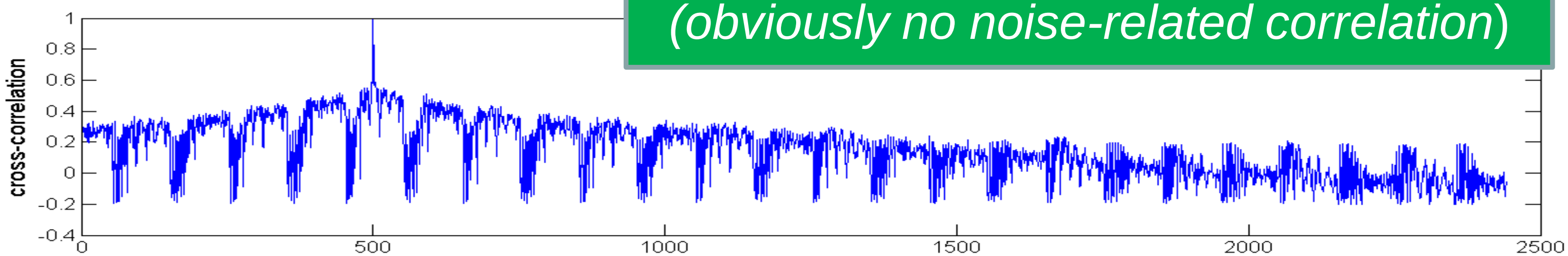
avg. trace – single trace

- Real traces



- Simulated traces

LOOKS GOOD
(obviously no noise-related correlation)



Is it enough?

- Sliding S + N simulator prevents the ρ distinguisher in contexts where noise-based correlation dominates
  - (& the signal is hard to exploit/hybridize)
  - ***Achievable for certain large // implementations***

Is it enough?

- Sliding S + N simulator prevents the ρ distinguisher in contexts where noise-based correlation dominates
  - (& the signal is hard to exploit/hybridize)
  - ***Achievable for certain large // implementations***

- ***Work in progress.*** Further investigations are needed
  - Maintain the signal variance (modified because of the sum in the sliding simulator): easy!
  - Different settings, simulators, designs, …

Is it enough?

- Sliding S + N simulator prevents the ρ distinguisher in contexts where noise-based correlation dominates
  - (& the signal is hard to exploit/hybridize)
  - ***Achievable for certain large // implementations***

- ***Work in progress.*** Further investigations are needed
  - Maintain the signal variance (modified because of the sum in the sliding simulator): easy!
  - Different settings, simulators, designs, …

Reminder: simulatability is the only empirically verifiable leakage assumption we currently have!

STAY TUNED

http://perso.uclouvain.be/fstandae/