Parallelizable (Authenticated) Online Ciphers

Elena Andreeva

COSIC, KU Leuven

Joint work with: A.Bogdanov, A.Luykx, B.Mennink, E.Tischhauser and K.Yasuda

DIAC, Chicago
13/08/2013

This Talk

Nonce misuse resistant AE

↓

Provably secure AE

Online AND Parallelizable AE

↓

Efficient AE

Achieving Privacy

- **We need**
  - A) Randomization,
  - B) Stateful algorithm, or
  - C) Nonce

Privacy with Nonces

- **Nonce use popular in AE**
- **Nonce:** unique non-repeating value
  - E.g. counter 1, 2, 3, …

- **Problems**
  - not always easy to implement
  - people **DO** reuse nonce
  - if repeated, then we lose all security

Nonce Misuse Resistance

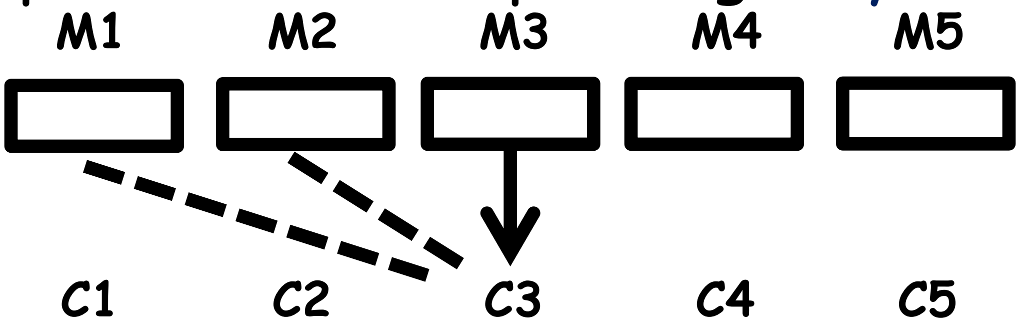
- **Misuse resistant AE**
  - if correct nonce use, then secure AE
  - else we still obtain **reasonable** security (no disaster even if nonce reused)

- **Examples of misuse resistant AE**
  1. SIV [RS06]: offline
  2. McOE [FFLW12]: authenticated online cipher

Online Ciphers for Misuse Resistance

- **Online cipher**

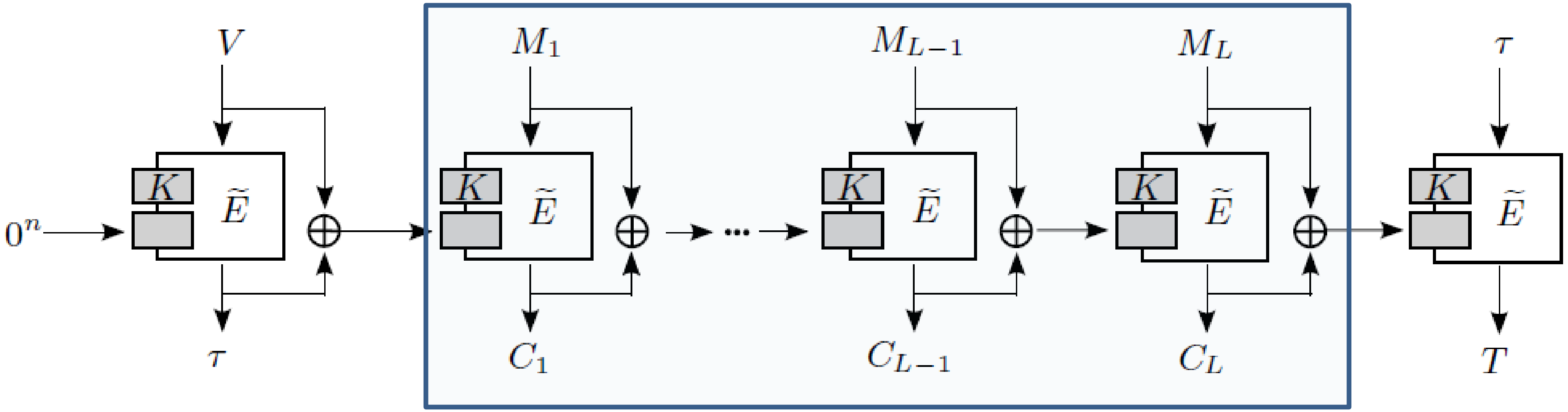
Cipher with Ci depending only on M1…Mi



- **No disaster**
  - If the 1st block is nonce, then perfect privacy
  - If not, then secure "up to common prefix"
- **Examples of online ciphers**
  - HCBC [BBKL01], M(H)CBC [N08], TC1/2/3 [RZ11]
- **Online cipher + Authentication → Authenticated Online Cipher**

Authenticated Online Cipher

- **McOE [FFLW12]**



- McOE-G: 1 BC + 1 multiplication in GF(2^n) per block
- completely sequential (Enc & Dec)
- adds authenticity to TC3 at minimal cost

(more efficient than generic composition)

Parallelizable Authenticated Online Cipher

- **Why?**
  - to improve efficiency

- **BUT** existing (authenticated) online ciphers are inherently sequential

- Intuitively, parallelizability appears difficult

How to Achieve Parallelizability?

- Do not feed ciphertext blocks into next block encryption

  ➔ use only **plaintext blocks** for “dependency”

- Plaintext under control of adversaries

  ➔ some “**masking**” required

Our Approach

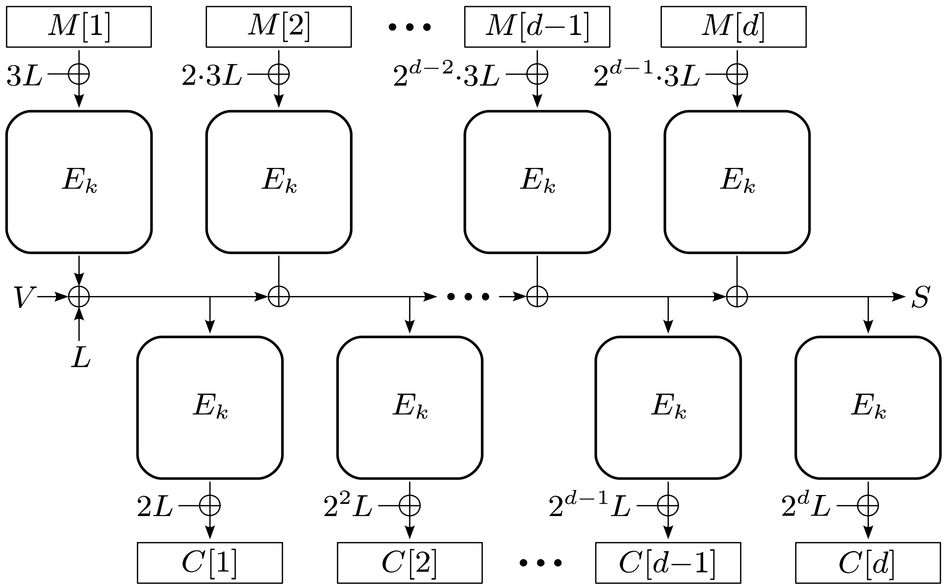
- **Design parallelizable online authenticated cipher in two stages:**
  1. Parallelizable online cipher (COPE)
  2. Dedicated authentication

↓

COPA

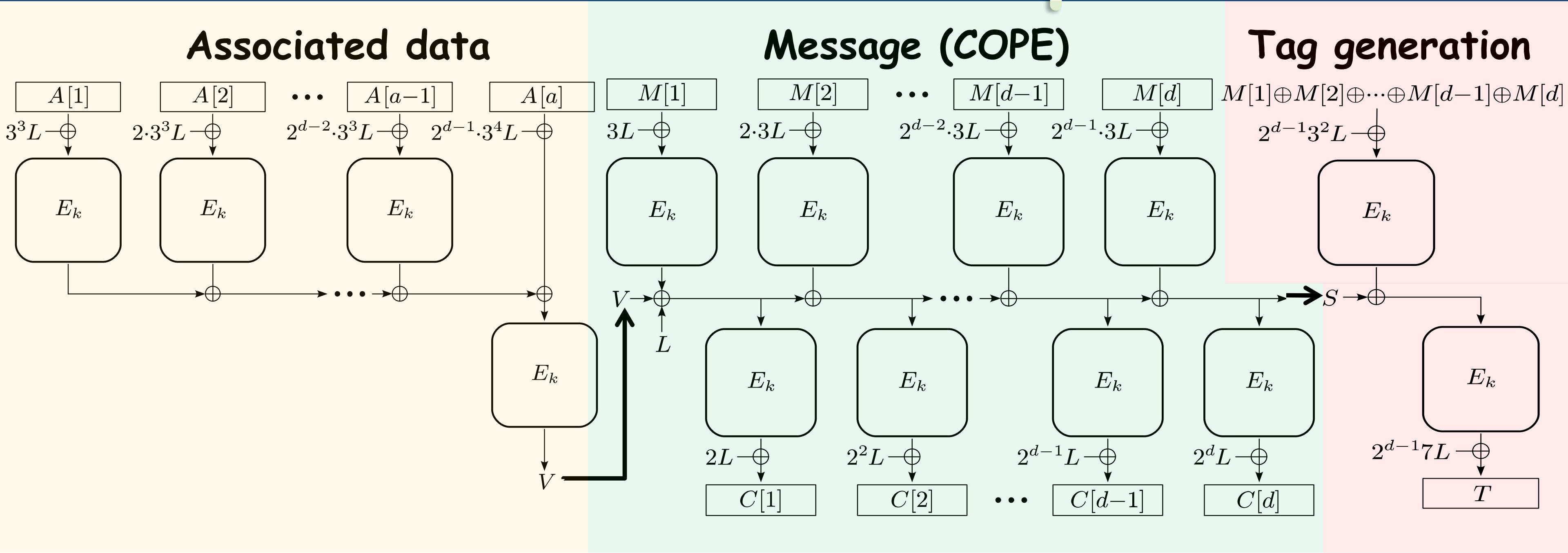
COPE: Parallelizable Online Cipher

$L = E_k(0)$



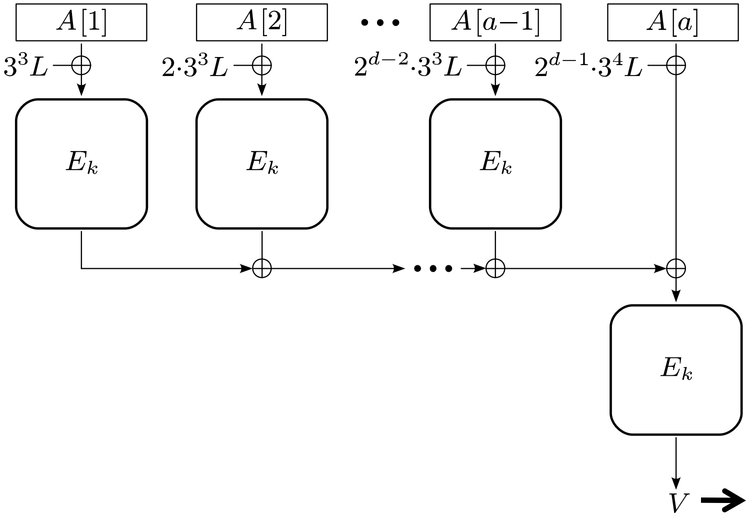
- Well parallelizable
- Single key + single primitive use
- 2 BC calls per block
- Online (nonce misuse resistant)
- Provably secure

COPA: Parallelizable Online Authenticated Cipher



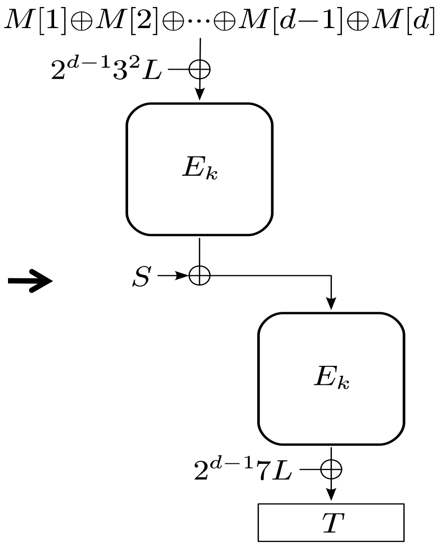
- Well parallelizable
- Single key + single primitive use
- 2 BC calls per block
- Online (nonce misuse resistant)
- Provably secure
- Dealing with fractional M: idea of XLS [RR07]

COPA: Processing Associated Data



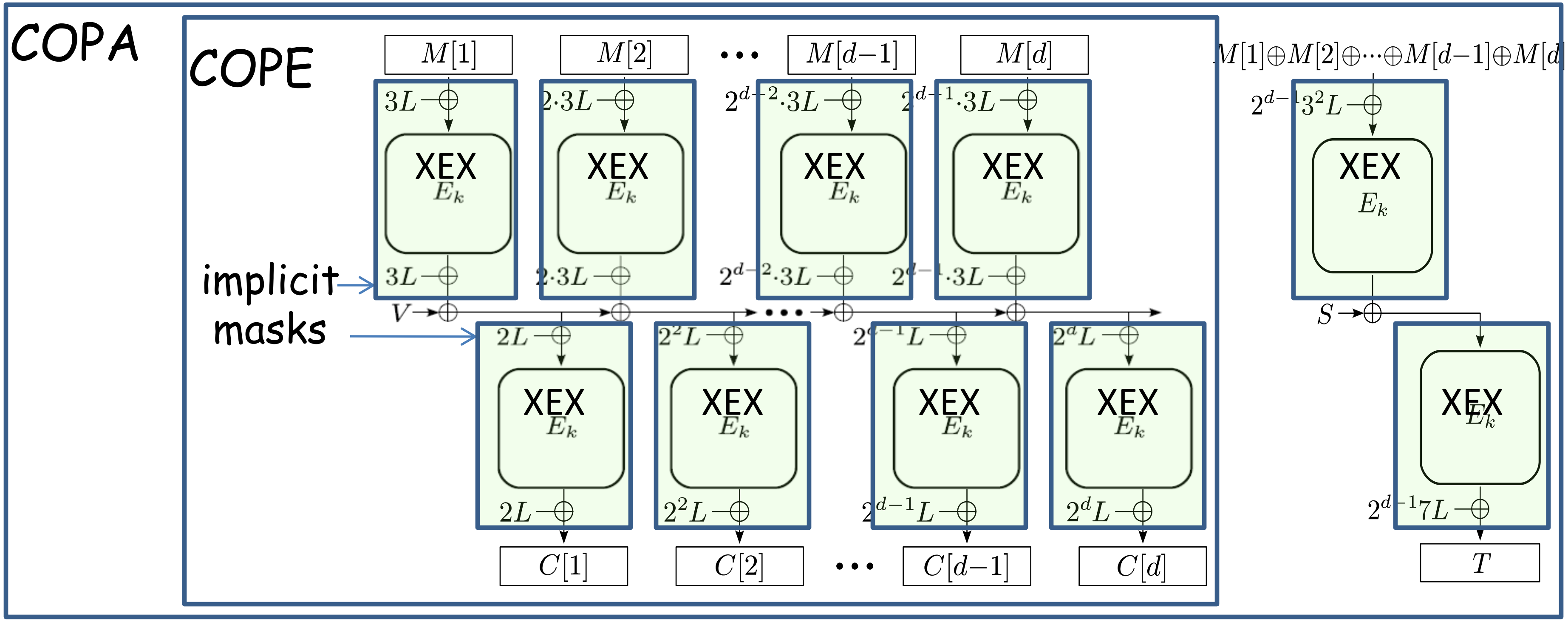
- **Well parallelizable**
- **1 BC call per AD block**

COPA: Tag Generation



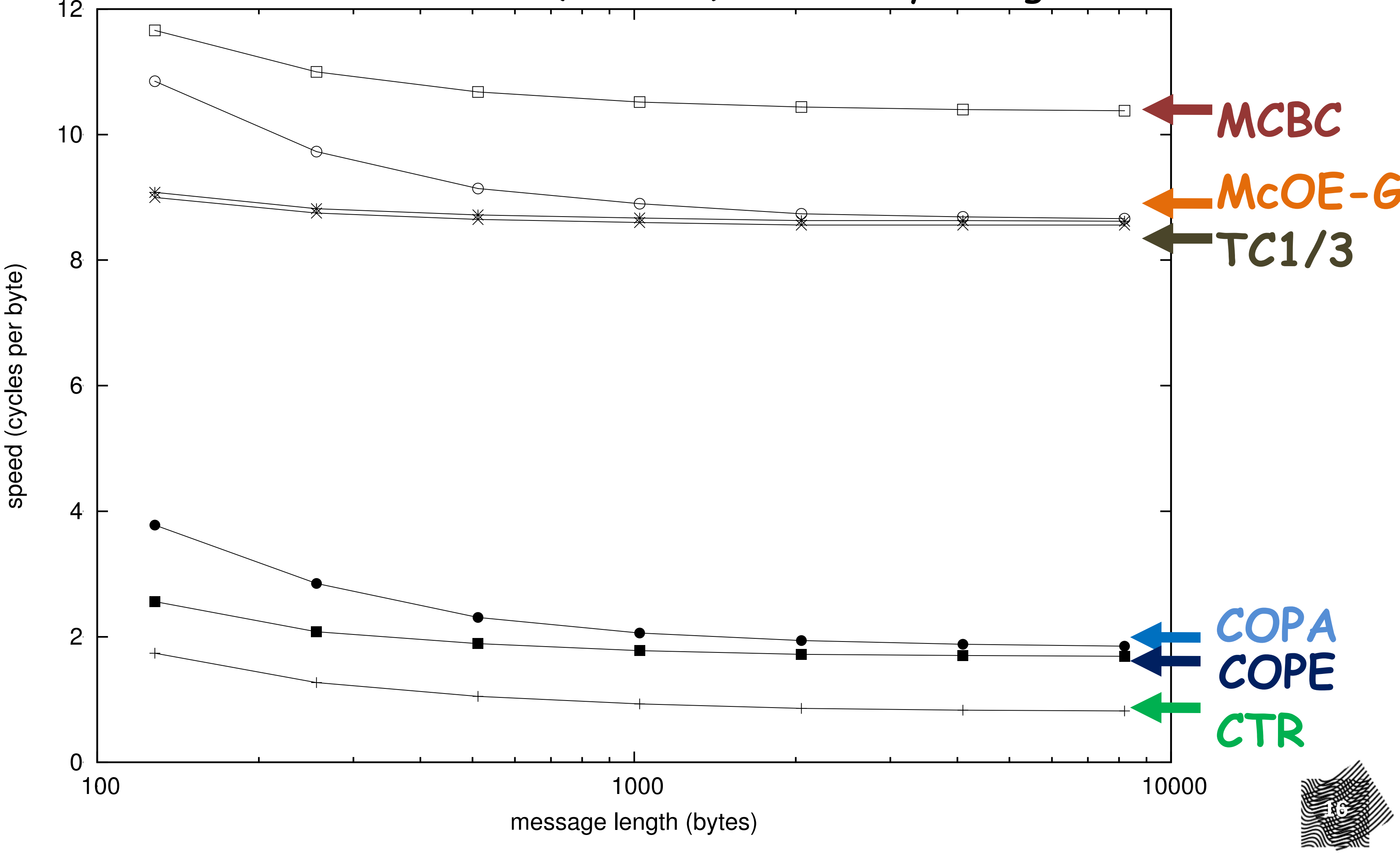
- Extends parallelizability of COPE
- 2 extra BC calls
- Online

Security



- 2 sequences of independent XEX evaluations
- Calculate the state collision probability (not trivial)
- **If E is SPRP, COPE is CPA secure up to $2^{n/2}$ queries**
- **If E is SPRP, COPA is AE secure up to $2^{n/2}$ queries**

COPE and COPA in Software



AES(128-bit)-NI Sandy Bridge

Summary

- COPE
  - parallelizable, online cipher
  - **5** times faster than TC1/3

- COPA = COPE + authentication
  - inherits COPE's properties
  - **5** times faster than McOE-G

Thank you!