



On Fast Algebraic Attacks

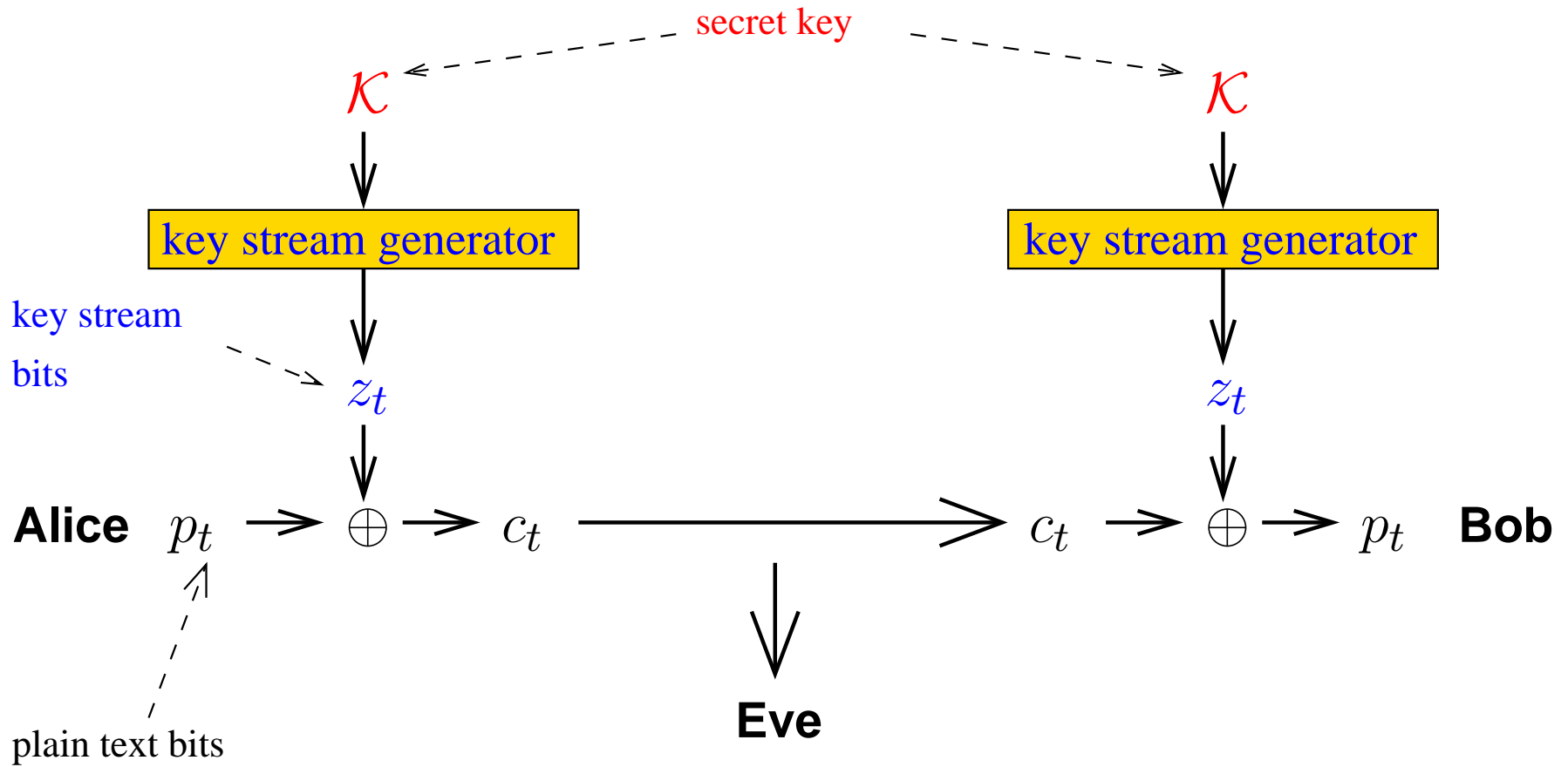
***Talk at the 9th Estonian Winter School in Computer Science, Palmse,
Estonia, March 4th 2004***

Frederik Armknecht

`armknecht@th.informatik.uni-mannheim.de`

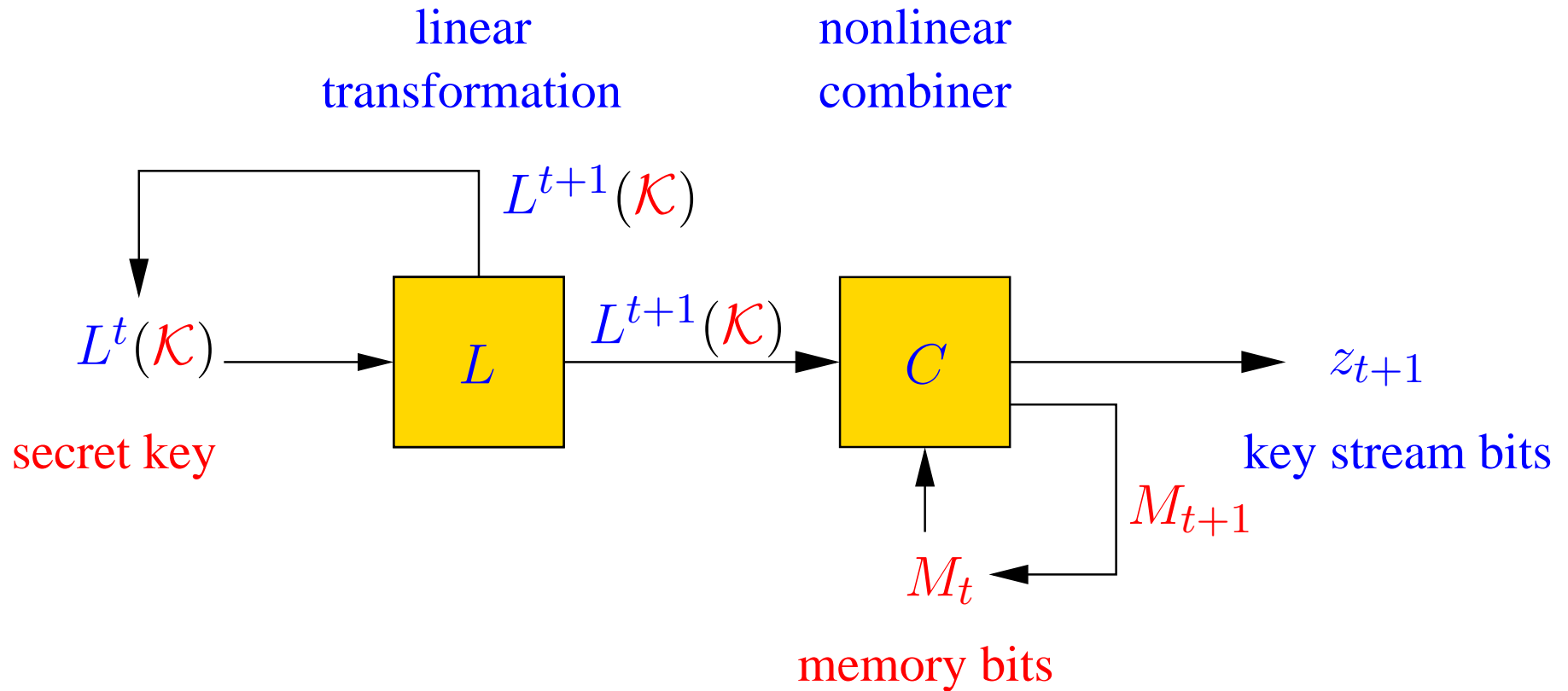
University of Mannheim
(Germany)

Key Stream Generator



LFSR-based key stream generators

Combiner with memory



Algebraic Attack - Basic Idea

1. Set up system of equations

2. Solve it

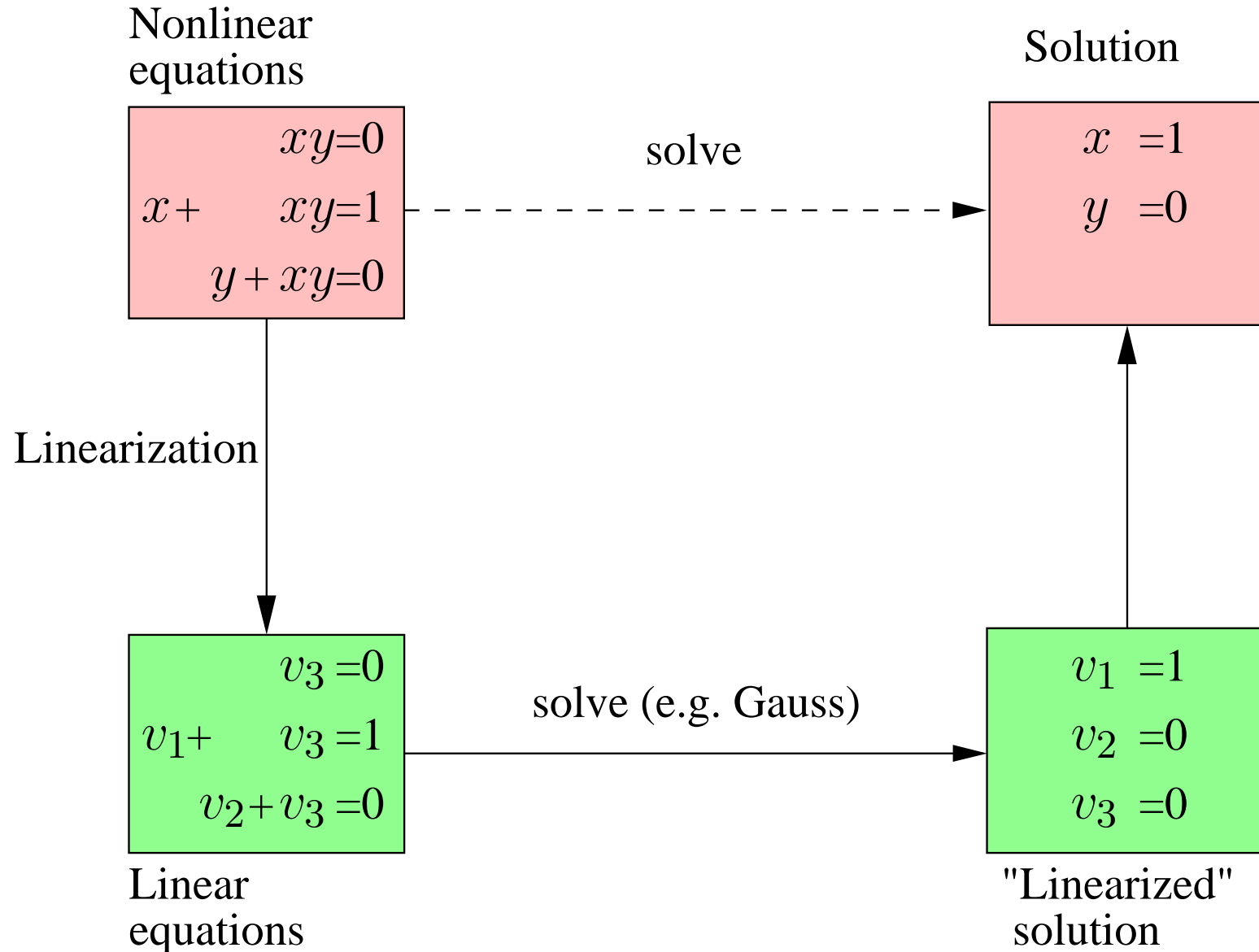
1. Ad-hoc equations

$$\begin{aligned}0 &= F(\mathcal{K}, \dots, L^{r-1}(\mathcal{K}), z_0, \dots, z_{r-1}) \\0 &= F(L(\mathcal{K}), \dots, L^r(\mathcal{K}), z_1, \dots, z_r) \\0 &= F(L^2(\mathcal{K}), \dots, L^{r+1}(\mathcal{K}), z_2, \dots, z_{r+1}) \\0 &= F(L^3(\mathcal{K}), \dots, L^{r+2}(\mathcal{K}), z_3, \dots, z_{r+2}) \\&\vdots\end{aligned}$$

Krause, Armknecht (Crypto '03):

- Exist always
- Can be found systematically

2. Linearization - Example



2. Linearization - Complexity

System of equations in $n := |\mathcal{K}|$ unknowns:

$$\begin{array}{c} \vdots \\ 0 = F(L^t(\mathcal{K}), \dots, L^{t+r-1}(\mathcal{K}), z_t, \dots, z_{t+r-1}) \\ \vdots \end{array}$$

$$\text{Degree} \leq d = \deg(F)$$

$$\Rightarrow \# \text{ Monomials} \leq \binom{n}{0} + \dots + \binom{n}{d} \in O(n^d)$$

$$\Rightarrow \text{Linearization} \quad O(n^{3d}) \text{ operations}$$

The lower the degree of F the better

Fast algebraic attacks

Courtois (Crypto '03):

- Motivation: New equations with lower degree

Fast algebraic attacks

Courtois (Crypto '03):

- Motivation: New equations with lower degree
- Precondition: Ad-hoc equation can be splitted

$$0 = F(L^t(\mathcal{K}), \dots, L^{t+r}(\mathcal{K}), z_t, \dots, z_{t+r})$$

$$= F_1(L^t(\mathcal{K}), \dots, L^{t+r}(\mathcal{K})) \quad (\text{high degree})$$

$$+ F_2(L^t(\mathcal{K}), \dots, L^{t+r}(\mathcal{K}), z_t, \dots, z_{t+r}) \quad (\text{low degree})$$

Fast algebraic attacks

Courtois (Crypto '03):

- Motivation: New equations with lower degree

- Precondition: Ad-hoc equation can be splitted

$$0 = F(L^t(\mathcal{K}), \dots, L^{t+r}(\mathcal{K}), z_t, \dots, z_{t+r})$$

$$= F_1(L^t(\mathcal{K}), \dots, L^{t+r}(\mathcal{K})) \quad (\text{high degree})$$

$$+ F_2(L^t(\mathcal{K}), \dots, L^{t+r}(\mathcal{K}), z_t, \dots, z_{t+r}) \quad (\text{low degree})$$

- Pre-computation: Reduce degree

Fastest attacks against ...

- Toyocrypt
- LILI-128
- E_0 (Bluetooth)
- Summation generator
- Sober-t32

1. System of equations

$$\begin{array}{lcl} F_1 (L^t(\mathcal{K}), \dots,) & + F_2 (L^t(\mathcal{K}), \dots, z_t, \dots) & = 0 \\ & \vdots & \\ F_1 (L^{t+R}(\mathcal{K}), \dots,) & + F_2 (L^{t+R}(\mathcal{K}), \dots, z_{t+R}, \dots) & = 0 \end{array}$$

Precomputation - 2

2. Find coefficients $c_0, \dots, c_R \dots$

$$\begin{array}{rcl} c_0 \cdot F_1(L^t(\mathcal{K}), \dots,) & + c_0 \cdot F_2(L^t(\mathcal{K}), \dots, z_t, \dots) & = 0 \\ & \vdots & \\ c_R \cdot F_1(L^{t+R}(\mathcal{K}), \dots,) & + c_R \cdot F_2(L^{t+R}(\mathcal{K}), \dots, z_{t+R}, \dots) & = 0 \end{array}$$

Precomputation - 2

2. Find coefficients $c_0, \dots, c_R \dots$

$$\begin{array}{rcl} c_0 \cdot F_1(L^t(\mathcal{K}), \dots,) & + c_0 \cdot F_2(L^t(\mathcal{K}), \dots, z_t, \dots) & = 0 \\ & \vdots & \\ c_R \cdot F_1(L^{t+R}(\mathcal{K}), \dots,) & + c_R \cdot F_2(L^{t+R}(\mathcal{K}), \dots, z_{t+R}, \dots) & = 0 \end{array}$$

\dots with $\sum c_i F_1(L^{t+i}(\mathcal{K}), \dots,) = 0$ for all t and \mathcal{K}

Precomputation - 2

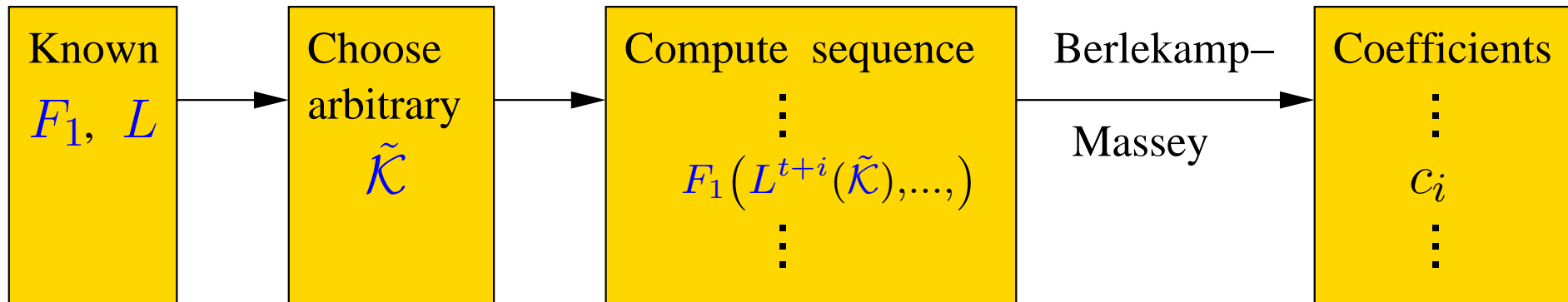
2. Find coefficients $c_0, \dots, c_R \dots$

$$\begin{array}{rcl} c_0 \cdot F_1(L^t(\mathcal{K}), \dots,) & + c_0 \cdot F_2(L^t(\mathcal{K}), \dots, z_t, \dots) & = 0 \\ & \vdots & \\ c_R \cdot F_1(L^{t+R}(\mathcal{K}), \dots,) & + c_R \cdot F_2(L^{t+R}(\mathcal{K}), \dots, z_{t+R}, \dots) & = 0 \end{array}$$

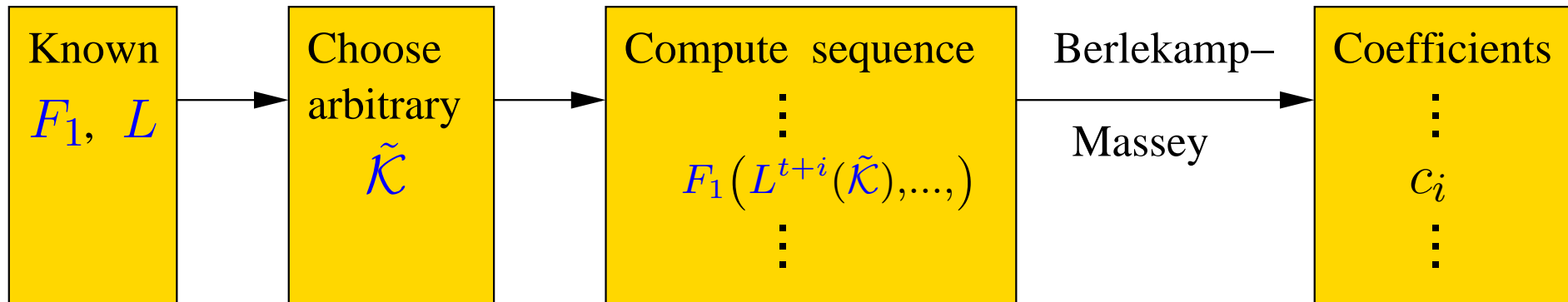
... with $\sum c_i F_1(L^{t+i}(\mathcal{K}), \dots,) = 0$ for all t and \mathcal{K}

$\Rightarrow \sum c_i F_2(L^{t+i}(\mathcal{K}), \dots, z_{t+i}, \dots) = 0$ with low degree

Finding c_0, \dots, c_R

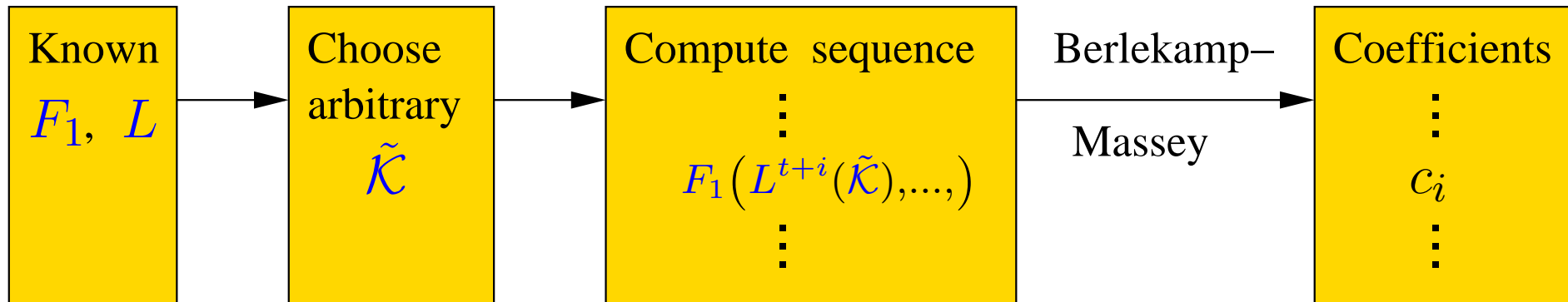


Finding c_0, \dots, c_R



Very fast !!!

Finding c_0, \dots, c_R



Very fast !!!

Correct ???

Proof of Correctness

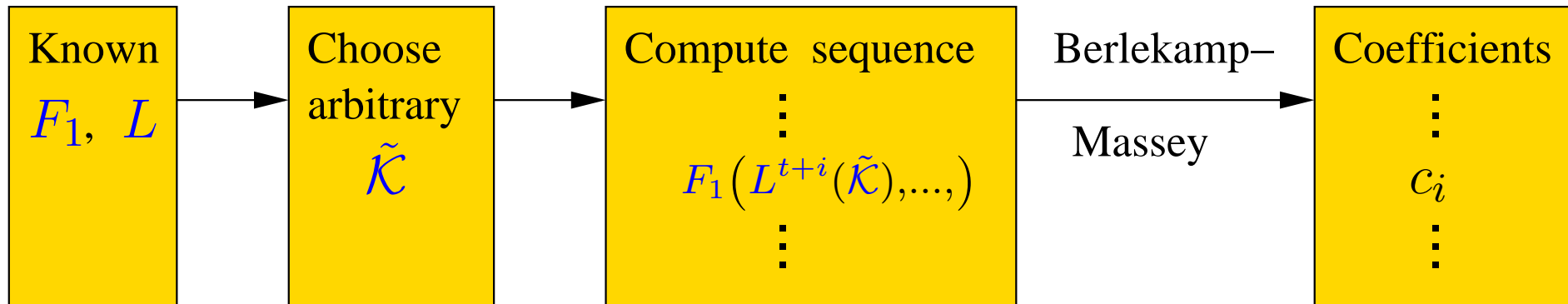
Assumption	Algorithm Correct?	Assumption true for E_0 ?
Strong	YES	NO
???	YES	YES
None	NO	YES

First Result:

- New Proof of Correctness
 1. Toyocrypt: OK
 2. LILI-128: OK
 3. E_0 (Bluetooth): OK
- Based on weaker assumption

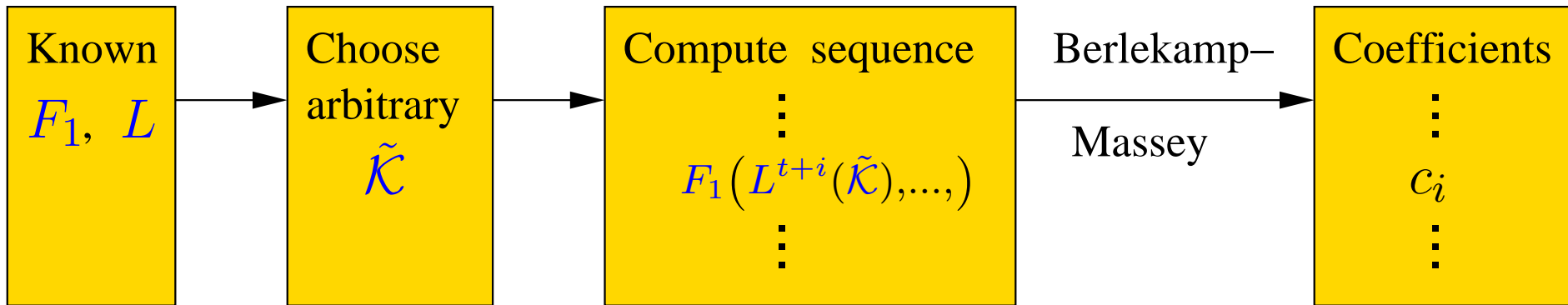
New precomputation step

Old precomputation step:

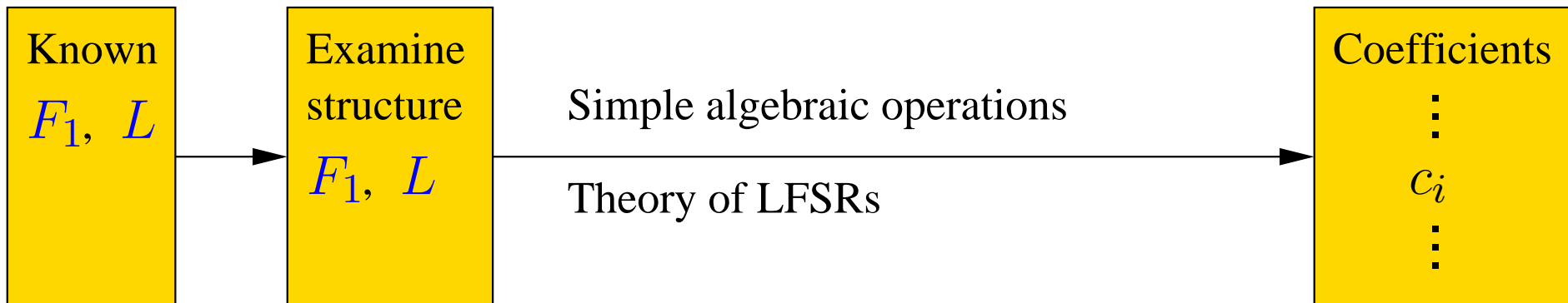


New precomputation step

Old precomputation step:



New precomputation step:



Second Result:

- New pre-computation step
 - ◆ Faster
 - ◆ Parallelizable

Computer Simulations

Reduced versions of E_0

Key size	Old				New			Old/New
19		10 h	41 m	43 s		12 m	03 s	53.32
19		11 h	02 m	49 s		12 m	07 s	54.75
19		10 h	50 m	00 s		11 m	59 s	54.30
19		10 h	52 m	59 s		11 m	55 s	54.86
19		10 h	53 m	31 s		11 m	58 s	54.65
23	3 d	06 h	30 m	16 s	1 h	43 m	25 s	45.55
25	18 d	18 h	26 m	00 s	13 h	50 m	07 s	32.56

- **New proof of correctness**

- ◆ Based on weaker assumption
- ◆ Shows correctness of fast algebraic attacks for E_0

- **New precomputation step**

- ◆ Faster
- ◆ Parallelizable