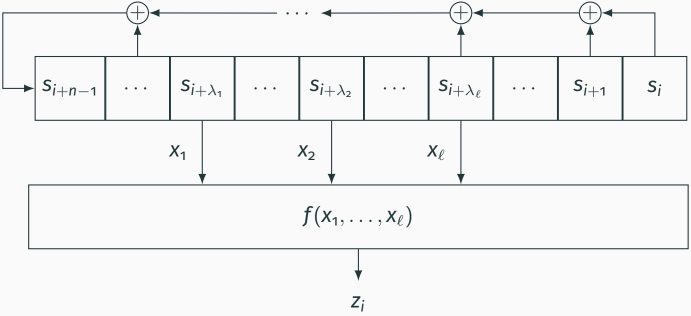
On the cryptanalysis of LFSR based stream ciphers

Isaac Canales-Martínez Igor Semaev

May ?, 2019

University of Bergen

Filter Generator - An LFSR based stream cipher



- Linear Feedback Shift Register (LFSR) of length n, defined by its degree-n feedback polynomial $g \in \mathbb{F}_2[x]$.
- $S_i := (s_i, \dots, s_{i+n-1})$ is the LFSR's state at time i.
- Filtering function $f : \mathbb{F}_2^\ell \to \mathbb{F}_2$.
- Inputs to f: values in cells with indices $\lambda_1, \dots, \lambda_\ell$.

Correlation and Fast Correlation Attacks

Correlation attack (CA):

- Published by Siegenthaler in 1985.
- Aimed at attacking combination generators (s LFSRs of length $n_i, i = 1, \ldots, s$.)

Fast Correlation Attack (FCA):

- Published by Meier and Staffelbach in 1989.
- The feedback polynomial g must have a low number of non-zero coefficients (< 10.)
- Use of low-density parity check equations.
- Use of *a priori* and *a posteriori* correlation probabilities for z_i.
- Complexity: $O(2^{cn})$ for some positive $c < 1$.

Attacks against the filter generator

- FCA-based:
  - State the problem as a decoding problem.
  - Improvements on finding parity checks (low-density multiples of g) and evaluating them.
  - Partial brute force.
  - Vectorial versions.
  - etc.
- Inversion attack / Filtering function oriented.
- Algebraic attacks
- ... and others.

Fast Correlation Attack

Let $z = z_1, \dots, z_N$ be the given key stream of length N.

A parity check equation (PCE) comes from congruences

$$1 + x^{i_1} + \dots + x^{i_{d-1}} \equiv 0 \bmod g.$$

for $0 < i_1 < \dots < i_{d-1} < N$. It only depends on g and N. If d is a small integer, we call it a low-density parity check equation (LDPCE).

For any j, we have that

$$S_j + S_{j+i_1} + \dots + S_{j+i_{d-1}} = 0.$$

Fast Correlation Attack

Let $p := \Pr(u(S) = f(S))$ be relatively high, for a linear function u.

Let $v_k := u(S_k) \oplus z_k$. Then $\Pr(v_k = 0) = p$ and a PCE implies

$$v_j + v_{j+i_1} + \cdots + v_{j+i_{d-1}} = z_j + z_{j+i_1} + \cdots + z_{j+i_{d-1}}. \tag{1}$$

Find many PCE and compute $p_k := \Pr\left(v_k = 0 \mid \text{relations (1)}\right)$.

Choose a set A s.t. for $k \in A$, $p_k \approx 1$. Get the initial state by solving

$$u(S_k) = z_k \oplus v_k, k \in A.$$

The method works for small d and moderate N. For larger d the complexity becomes exponential.

Some additional notation...

- Let M be the companion matrix of g, then

$$S_i = M^i S_0.$$

- Let Λ denote the $\ell \times n$ matrix that "selects" the inputs to f:

$$(s_{i+\lambda_1}, s_{i+\lambda_2}, \ldots, s_{i+\lambda_\ell}) = \Lambda S_i.$$

- Define $A_i := \Lambda M^i$, an $\ell \times n$ matrix of rank ℓ.
- Let $X = S_0$, then

$$z_i = f(A_i X).$$

Statement of the problem

Given N bits of the key stream, determine the vector a for which the conditional probability

$$\mathbf{Pr}\left(X = a \mid f(A_iX) = z_i, i = 1, \ldots, N\right).$$

is maximal.

Straightforward method:

Equivalent to solving the system of equations
$f(A_iX) = z_i, i = 1, \ldots, N$ ... but may be computationally infeasible.

Our approach

- Compute the conditional distributions

$$\mathbf{Pr}\left(BX = b \,\middle|\, \begin{matrix} f(A_{i_1}X) = z_{i_1} \\ \vdots \\ f(A_{i_d}X) = z_{i_d} \end{matrix} \right)$$

  for a variety of matrices B of different ranks and indices $\{i_1, \ldots, i_d\}$, where d is small.
- Each B above define a *level*. At each level we combine the computed distributions with a maximum likelihood (ML) method to get the initial state X.

Generalised LDPCE

A Generalised LDPCE comes from

$$h_0 + x^{i_1}h_1 + \cdots + x^{i_{d-1}}h_{d-1} \equiv 0 \bmod g,$$

where the polynomials h_i are in the space generated by $x^{\lambda_1}, \ldots, x^{\lambda_\ell}$.

Why Generalised LDPCE?

- Average key stream length to find LDPCE: $N > d\,2^{\frac{n}{d-1}}$.
- Average key stream length to find Generalised LDPCE: $N > d\,2^{\frac{n-d\ell}{d-1}}$.

Short relations

Generalised LDPCE are a particular case of the relations

$$c_{i_1}A_{i_1} + \cdots + c_{i_d}A_{i_d} \in \langle B \rangle.$$

These relations are called short if d is small.

We use short relations to determine the distributions to compute

$$\mathbf{Pr}\left(BX = b \,\middle|\, \begin{matrix} f(A_{i_1}X) = z_{i_1} \\ \vdots \\ f(A_{i_d}X) = z_{i_d} \end{matrix} \right).$$

Getting the initial state

Distinguish $X = S_0$ from random X:

- β desired success probability.
- For each level B ($r \times n$ matrix), distinguish $b = BX$:
  - set threshold c_β.
  - compare ML indicator of b with c_β.
  - if passed, extend b to the next level.

Expected number of survivors:

- The average number of survivors at each level is $\alpha 2^r$, where α depends on the ML indicator.
- We use a multivariate normal approximation to compute α.

Some (small) experimental results

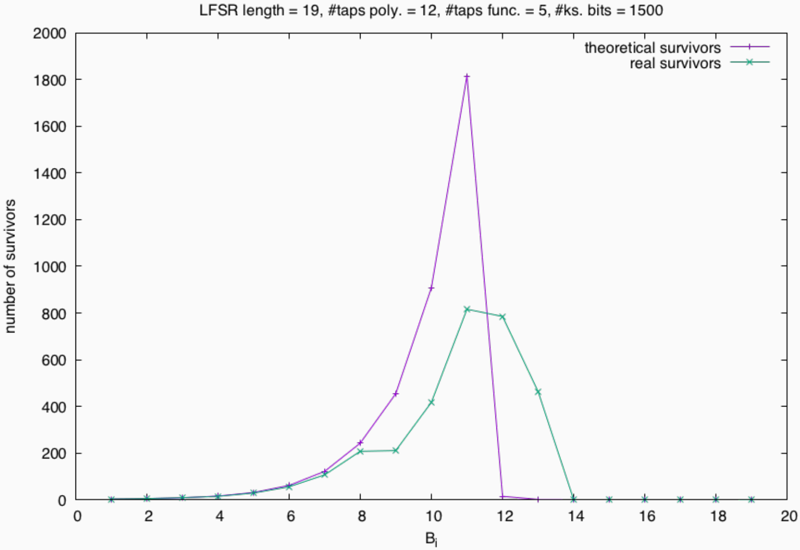
Device:

- $g = x^{19} + x^{16} + x^{14} + x^{13} + x^{10} + x^9 + x^8 + x^7 + x^6 + x^5 + x^4 + x^2 + 1$.
- $f = x_1x_2x_3 + x_1x_2x_4 + x_1x_2x_5 + x_1x_4 + x_2x_5 + x_3 + x_4 + x_5$.
- $(\lambda_5, \ldots, \lambda_1) = (18, 16, 13, 9, 1)$.

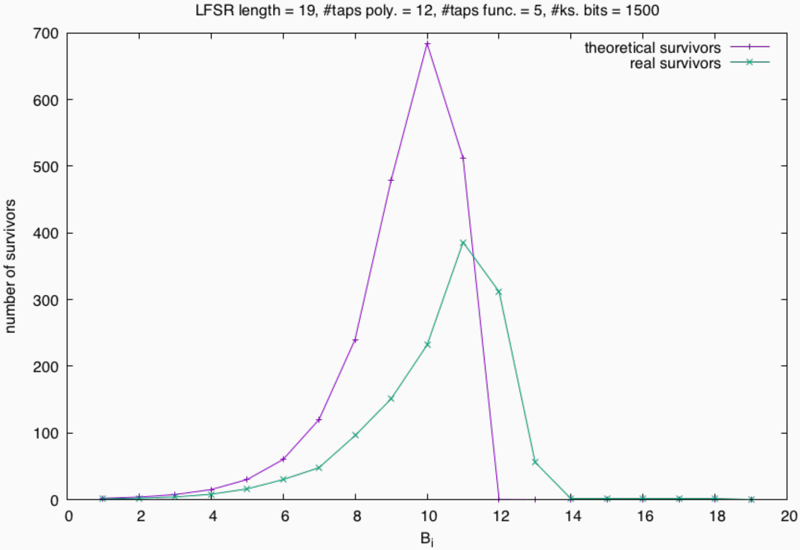
It is a “hard” instance for the given parameters:

- high number of non-zero coefficients in g, and
- $\lambda_5 - \lambda_1 \approx n$.

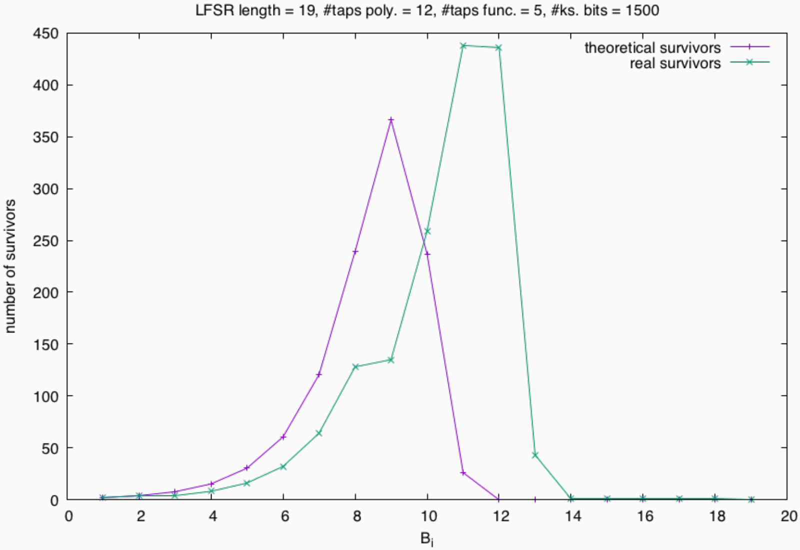
Some (small) experimental results



Some (small) experimental results



Some (small) experimental results



Questions?