An Algorithm for the Initial State Reconstruction of the Clock–Controlled Shift Register

Miodrag V. Živković *

Abstract

An algorithm is given for the reconstruction of the initial state of a key–stream generator (KSG) consisting of a short linear feedback shift register (length ≤ 30), whose clock is controlled by an algebraically simple internal KSG. The algorithm is based on the fact that the expected number of possible LFSR initial states exponentially decreases with the length of the known part of the output sequence.

Index Terms — Clock–controlled FSR (feedback shift register), Levenshtein distance.

*The author is with the Institute of Applied Mathematics and Electronics, Belgrade, Yugoslavia

Denote by G the key–stream generator (KSG), consisting of an internal KSG G_0, and a binary linear feedback shift–register (LFSR), whose clock is controlled by G_0, see [1], [2]. In this paper the problem of determining the initial state of KSG G, given its output sequence, is considered. Denote the binary output sequences of KSG G_0, LFSR, and KSG G by $a_1, a_2, \ldots, a_n, \ldots$; $b_1, b_2, \ldots, b_n, \ldots$ and $c_1, c_2, \ldots, c_n, \ldots$, respectively. The output sequence of the LFSR is defined by the values $b_1, b_2, \ldots, b_k$, and by the recurrence relation

$$b_n = \bigoplus_{i=1}^{k} h_i b_{n-i}, \qquad n > k, \tag{1}$$

of order k over the field GF(2). Here k is the length of the LFSR, and $h_1, h_2, \ldots, h_k \in \mathrm{GF}(2)$ are its feedback coefficients. The output sequence from G is obtained by the decimation of the output sequence from the LFSR,

$$c_n = b_{r_n}, \qquad n \geq 1, \tag{2}$$

where $r_n = n + \sum_{i=1}^{n} a_i, \quad n \geq 1$, or equivalently

$$r_{n+1} = r_n + 1 + a_{n+1}, \qquad n \geq 1. \tag{3}$$

The general case is very complicated, so we start with the following assumptions:

- the first N members $c_1, c_2, \ldots, c_N$ of the output sequence from G are known, $N > 0$;
- the length of the LFSR is not too large (for example $k \leq 30$);
- knowing $K > 0$ arbitrary members of the sequence $a_1, a_2, \ldots, a_n, \ldots$, one can effectively determine the initial state of G_0 (G_0 could be another LFSR, for example).

The main part of the reconstruction algorithm is to find the actual positions of some output bits over a period of the LFSR output sequence. For that reason we first define an embedding relation between binary vectors.

Definition 1 *Suppose* $\underset{\sim}{\mathbf{b}} = (b_1, b_2, \ldots, b_B)$ *and* $\underset{\sim}{\mathbf{c}} = (c_1, c_2, \ldots, c_N)$ *are binary vectors of length* B *and* N *respectively,* $B, N > 0$. *We say that* $\underset{\sim}{\mathbf{c}}$

can be embedded into $\underset{\sim}{\mathbf{b}}$ *if there exist integers* $z_0 = 0, z_1, \ldots, z_N$ *such that*

$$c_i = b_{z_i},\ z_i - z_{i-1} \in \{1, 2\}, \qquad 1 \le i \le N.$$

If $z_N = B$ *then* $\underset{\sim}{\mathbf{c}}$ *can be embedded* onto $\underset{\sim}{\mathbf{b}}$.

Obviously, $\underset{\sim}{\mathbf{c}}$ can be embedded into $\underset{\sim}{\mathbf{b}}$ if the sequence starting with $\underset{\sim}{\mathbf{c}}$, can be obtained by decimating the sequence starting with $\underset{\sim}{\mathbf{b}}$ under the control of the sequence $z_0, z_1, \ldots$. The following theorem describes the behavior of the probability that the fixed vector can be embedded into a random binary vector, when the lengths of these vectors grow.

Theorem 1 *Let* $\underset{\sim}{\mathbf{c}} = (c_1, c_2, \ldots, c_N)$ *be an arbitrary binary vector of length* N*, and let* $\underset{\sim}{\mathbf{B}} = (B_1, B_2, \ldots, B_{2N})$ *be an* $2N$*–dimensional binary random variable with independent coordinates and uniform probability distribution. If* $P_{N,\,\underset{\sim}{\mathbf{c}}}$ *denotes the probability that vector* $\underset{\sim}{\mathbf{c}}$ *can be embedded into the random vector* $\underset{\sim}{\mathbf{B}}$ *and* $P_N = \min_{\underset{\sim}{\mathbf{c}}} P_{N,\,\underset{\sim}{\mathbf{c}}}$ *, then there exist* $\alpha, \beta > 0$*,* $\alpha < 1$*, such that*

$$P_N < \beta\alpha^N. \qquad \square \tag{4}$$

Proof. Our goal is to find an upper bound for the number of different vectors of length $2N$ into which the vector $\underset{\sim}{\mathbf{c}}$ can be embedded. Suppose $N = mM$, where m and M are integers. For $0 \le l \le m$ denote by $U_{m,l}$ the maximum number of different vectors of length $m+l$ into which an arbitrary

m	0	1	2	3	4	5	6	7	α_m	$-1/\log_2 \alpha_m$	p_m
2	1	3	4						0.9354143	10.3817861	0.0123972
3	1	4	8	8					0.9085603	7.2282625	0.0194109
4	1	5	13	20	16				0.8946455	6.2261849	0.0235022
5	1	6	19	38	48	32			0.8862936	5.7423940	0.0260349
6	1	7	26	63	104	112	64		0.8807541	5.4588471	0.0277883
7	1	8	34	96	192	272	256	128	0.8768163	5.2727731	0.0291521

Table 1: Quantities $U_{l,m}$, α_m and p_m for $2 \leq m \leq 7$

m–tuple can be embedded by inserting l bits (at most one after each member of the m–tuple). The numbers $U_{m,l}$ are given in Table 1 for $2 \leq m \leq 7$. The complexity of computing the numbers $U_{m,l}$, $0 \leq l \leq m$, by direct counting is $O\left(\sum_{l=0}^{m} 2^{m+l}\binom{m}{l}\right) = O(6^m)$. In Table 1 the numbers α_m and p_m are also given, where

$$\alpha_m = \frac{1}{2}\left(\sum_{l=0}^{m} U_{m,l} 2^{-l}\right)^{1/m}, \tag{5}$$

and p_m is the largest p from $[0, 1/2)$ such that

$$H(p) = -p\log_2 p - (1-p)\log_2(1-p) \leq -\log_2 \alpha_m.$$

For $m \geq 2$ we have $U_{0,m} = 1$ and $U_{m,m} = 2^m$. From the obvious inequality

$$U_{m,l} \leq \binom{m}{l} 2^l, \qquad 0 \leq l \leq m, \tag{6}$$

and the equality $U_{m,1} = m+1$ (which can be proved easily) it follows $\alpha_m < 1$, because in (6) for $l = 1$ the strict inequality holds.

Consider the set $\Phi(\underset{\sim}{\mathbf{c}})$ of vectors of length $2N$ into which $\underset{\sim}{\mathbf{c}}$ can be embedded. Obviously, $\underset{\sim}{\mathbf{b}} \in \Phi(\underset{\sim}{\mathbf{c}})$ if, and only if, $\underset{\sim}{\mathbf{b}}$ can be partitioned into $M+1$ parts, so that for $1 \leq i \leq M$ the vector $\underset{\sim}{\mathbf{c}}^{\,i} = (c_{mi-m+1}, \ldots, c_{mi})$ can be embedded onto i-th part of $\underset{\sim}{\mathbf{b}}$. Suppose that we have fixed the

number of insertions into $\underset{\sim}{\mathbf{c}}^{\,i}$, $1 \le i \le M$, and that I_l is the number of m–tuples $\underset{\sim}{\mathbf{c}}^{\,i}$ with l insertions, $0 \le l \le m$. Then the number of such vectors $\underset{\sim}{\mathbf{b}}$ is at most

$$\left(\prod_{l=0}^{m} U_l^{I_l}\right) \exp_2\left(2N - \sum_{l=0}^{m}(m+l)I_l\right) = 2^N \prod_{l=0}^{m}\left(U_l 2^{-l}\right)^{I_l},$$

because $\sum_{l=0}^{m} I_l = M$ and the last $2N - \sum_{l=0}^{m}(m+l)I_l$ bits of $\underset{\sim}{\mathbf{b}}$ can take the values from the set $\{0,1\}$ independently. For the fixed $I_0, I_1, \ldots, I_l$ we have $M!/(I_0!I_1!\cdots I_l!)$ possibilities to chose the numbers of inserted bits into the m–tuples $\underset{\sim}{\mathbf{c}}^{\,i}$, $1 \le i \le M$. Summing over the partitions set of M,

$$\left\{(I_0, I_1, \ldots, I_l) \mid I_0, I_1, \ldots, I_l \ge 0,\ \sum_{l=0}^{m} I_l = M\right\},$$

we get the upper bound on the cardinal number of $\Phi(\ \underset{\sim}{\mathbf{c}}\ )$,

$$\mid \Phi(\ \underset{\sim}{\mathbf{c}}\ ) \mid \le 2^N \sum \frac{M!}{I_0!I_1!\cdots I_l!} \prod_{l=0}^{m}\left(U_l 2^{-l}\right)^{I_l} = 2^{2N}\alpha_m^N,$$

and therefore (for $N = mM$) $P_N \le \alpha_m^N$. If $\underset{\sim}{\mathbf{c}}'$ is the vector obtained by inserting an arbitrary bit at the beginning of $\underset{\sim}{\mathbf{c}}$, then

$$\mid \Phi(\ \underset{\sim}{\mathbf{c}}'\ ) \mid \le 4 \mid \Phi(\ \underset{\sim}{\mathbf{c}}\ ) \mid \le 2^{2N+2} P_N,$$

which implies $P_{N+1} \leq P_N$. Combining this inequality with the previous one, we get the inequality $P_N \leq \alpha_m^{m[N/m]} \leq \alpha_m^{N-m+1}$, which is equivalent to (4) if $\alpha = \alpha_m$ and $\beta = \alpha_m^{1-m}$, $m \geq 2$. Here $[x]$ denotes the largest integer not greater than x. The values of α_m, $2 \leq m \leq 7$ can be found in Table 1. As they are decreasing with m (at least for $m \leq 7$), asymptotically best upper bound for P_N (if we restrict ourselves to the values of α_m from Table 1) is obtained for $m = 7$, $P_N \leq 0.8768^{7[N/7]}$. $\square$

The algorithm for determining the initial state of KSG G is based on the idea of finding sub–vectors over a period of the LFSR output sequence, into which the vectors $(c_i, c_{i+1}, \ldots, c_N)$, $1 \leq i < N$, can be embedded. As usual, $\delta_{i,j}$ denotes the Kronecker symbol,

$$\delta_{i,j} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} .$$

Algorithm 1 *The reconstruction of the initial state of the KSG G, given the part $\underset{\sim}{\mathbf{c}} = (c_1, c_2, \ldots, c_N)$ of its output sequence.*

1. *Let $B = P + N$, where P denotes the period of the LFSR output sequence (we assume that the LFSR generates the maximum length sequence; in other cases every cycle of the output sequence must be considered separately). Calculate the vector $\underset{\sim}{\mathbf{b}} = (b_1, b_2, \ldots, b_B)$, the part of the LFSR output sequence, where $b_1 = b_2 = \cdots = b_k = 1$, see (1).*

2. *Set $i \leftarrow N$, $t \leftarrow 0$, and for $1 \leq j \leq B$ set*

$$p_j \leftarrow \delta_{c_N, b_j}$$

(vector $\underset{\sim}{\mathbf{p}}$ contains the information on the possible positions of c_i in $\underset{\sim}{\mathbf{b}}$; i and t are counters).

3. *[Decrement i.] Set $i \leftarrow i - 1$; if $i = 0$ then go to 7.*

4. *For $1 \leq j \leq B$ if*

$$c_i = b_j \text{ and } ((j + 1 \leq B \text{ and } p_{j+1} = 1) \text{ or } (j + 2 \leq B \text{ and } p_{j+2} = 1))$$

then set $q_j \leftarrow 1$, *otherwise set* $q_j \leftarrow 0$ *(vector* $\underset{\sim}{\mathbf{q}}$ *contains the information on the possible positions of* c_i *in* $\underset{\sim}{\mathbf{b}}$ *).*

5. *[Is exactly one coordinate of* $\underset{\sim}{\mathbf{q}}$ *equal to* 1*?] If* $\sum_{i=1}^{B} q_j = 1$ *and* $q_s = 1$ *then set* $t \leftarrow t+1$, $u_t \leftarrow i$ *and* $v_t \leftarrow s$ *(in that case the bit* c_i *is obtained from the member* b_s *of the output sequence from* G *and* $r_i = s$, *see (2)).*

6. *For* $1 \le j \le B$ *set* $p_j \leftarrow q_j$ *and go to* 3.

7. *For every pair* $(j, j+1)$, $1 \le j < t$, *such that* $u_j = t$ *and* $u_{j+1} = t+1$, *compute* $a_{t+1} = v_{j+1} - v_j - 1$, *the member of the output sequence from* G_0, *see (3). If the number of such pairs is large enough, determine the initial state of* G_0 *(solving the appropriate set of equations, which is not hard, according to our assumptions).*

8. *Knowing the initial state of* G_0, *determine the initial state of the LFSR by solving the system of linear equations. There is also another possibility: the initial state of the LFSR is determined by the* k *subsequent bits from* $\underset{\sim}{\mathbf{b}}$ *, starting from some place where the corresponding coordinate of* $\underset{\sim}{\mathbf{p}}$ *is*

1. *According to Theorem 1, the number of such places is small if* N *is large enough.*

The main part of this algorithm is similar to the algorithm for computing the constrained Levenshtein distance [3, pp 286] between the vectors $\underset{\sim}{\mathbf{c}}$ and $\underset{\sim}{\mathbf{b}}$. Transforming the string $\underset{\sim}{\mathbf{c}}$ into the string $\underset{\sim}{\mathbf{b}}$, deletions, alterations and two subsequent insertions are not allowed, with the exception of an arbitrary number of insertions at the beginning and at the end of $\underset{\sim}{\mathbf{c}}$.

By induction over i, $i = N, N-1, \ldots, 1$, it can be proved that in the $(N-i+1)$–th step we have $q(t) = 1$ if, and only if, the sequence $(c_i, c_{i+1}, \ldots)$ can be embedded into $(b_t, b_{t+1}, \ldots, b_B)$ with no insertions at the beginning.

According to Theorem 1, the probability of "placing" the sequence $\underset{\sim}{\mathbf{c}}$ in wrong places inside the sequence $\underset{\sim}{\mathbf{b}}$ is small for N large enough.

Using Theorem 1 it is possible to estimate the necessary length N of the output sequence. The probability that $\underset{\sim}{\mathbf{c}}$ cannot be embedded in any (wrong) position over a period of the LFSR output sequence is lower–bounded approximately (if we neglect the dependence between the possible embeddings) by $\left(1-\beta\alpha^N\right)^{K/2}$, where $K=2^k$. If $\beta\alpha^N \times K/2 \ll 1$, then this bound is greater than $1/2$ if approximately $N > -(k+\beta)/\log_2 \alpha \simeq -k/\log_2 \alpha$. Note that this inequality implies that the previous assumption is satisfied. For $\alpha = \alpha_7$ we get the condition $n > 5.3k$, see Table 1. Thus, for the reconstruction of the LFSR initial state we need the output sequence which is approximately 5 times longer than the LFSR. Of course, it might be necessary to have a longer part of the output sequence to determine the initial state of G_0.

The main limitation of Algorithm 1 arises from its numerical complexity, which is of order $N2^k$. The reconstruction problem cannot be solved effectively by Algorithm 1 if the length k of the LFSR is greater than 30, for example.

Theorem 1 can be extended to the more complicated statistical model of the KSG, obtained by adding modulo 2 the independent noise of probability p, $p < 1/2$, to the output of G. Then we can consider the string $\underset{\sim}{\mathbf{c}}'$, obtained from the string $\underset{\sim}{\mathbf{c}}$ by inserting independent errors with probability p. The number of such strings (they differ from $\underset{\sim}{\mathbf{c}}$ in approximately pN places) is upper–bounded approximately by $2^{NH(p)}$. The number of different vectors of length $2N$ into which some corrupted versions of $\underset{\sim}{\mathbf{c}}$ can be embedded is not

greater than

$$2^{NH(p)}2^{2N}\beta\alpha^N = \beta \exp_2 \left(N(2 + H(p) + \log_2 \alpha)\right),$$

and the probability to pick at random some of them is bounded approximately by $\beta \exp_2 (N(H(p) + \log_2 \alpha))$. This probability tends to zero when $N \to \infty$ only if $H(p) < -\log_2 \alpha$, or $p < 0.0292$ for $\alpha = \alpha_7$. In that case it is possible to reduce the number of solutions for the initial state of the LFSR, but there still remains the problem of effectively determining the initial states of other parts of the KSG.

Acknowledgement

The author is grateful to anonymous referees for useful remarks, and also for the suggestion to improve the upper bound in Theorem 1.

References

[1] W. G. Chambers, S. M. Jennings, "Linear equivalence of certain BRM shift–register sequences" *Electronics Letters*, vol. 20, pp. 1018–1019, 1984.

[2] D. Gollmann, W. G. Chambers, "Clock–controlled shift registers: A review", *IEEE Journal on Selected Areas in Communications*, vol. SAC–7, pp. 525–533, 1989.

[3] D. Sankof, J. B. Kruskal, *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Reading, MA: Addison–Wesley, 1983.