

Chiffrements symétriques

Laurent Fousse

laurent.fousse@imag.fr

19 mars 2010

Plan

1 Linear Feedback Shift Register

Plan

1 Linear Feedback Shift Register

Linear Feedback Shift Registers (LFSR)

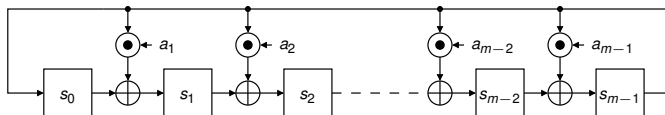
- Un algorithme pour produire un flux de bits.
- Utilisé comme un générateur de nombre aléatoire dans le cadre des chiffrements par flot additifs.

Exemple d'utilisation: E0

E0 est un chiffrement par flot utilisé pour les communications Bluetooth.

- Chiffrement par flot binaire additif
- 4 LFSR de tailles 25, 31, 33 and 39 bits (total: 128).
- 4 bits d'état interne.

Linear Feedback Shift Registers (Galois setup)



$$s'_0 = s_{m-1}$$

$$s'_i = s_{i-1} + s_{m-1} a_i \text{ for } 0 < i < m$$

Linear Feedback Shift Registers (Galois setup)

Propriété du polynôme de rétroaction

$$P(X) = 1 + a_1X + a_2X^2 + \dots + a_{m-1}X^{m-1} + X^m$$

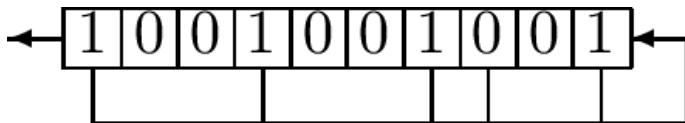
$$U(X) = s_0 + s_1X + \dots + s_{m-1}X^{m-1}$$

$$U'(X) = X \cdot U(X) \text{ mod } P(X).$$

Linear Feedback Shift Registers (Fibonacci setup)

- L registres numérotés $(0, 1, \dots, L - 1)$
- à chaque clock d'horloge, le contenu du registre 0 est copié dans la séquence de sortie.
- le contenu du registre i est déplacé dans le registre $i - 1$ pour $1 \leq i \leq L - 1$.
- le nouveau contenu du registre $L - 1$ est le bit de rétroaction, calculé comme l'addition (mod 2) du contenu précédent d'un certain ensemble fixé de registres.

Linear Feedback Shift Registers (Fibonacci setup)



$(s_0, s_1, \dots, s_{L-1})$ état initial

$$s_j = (c_1 s_{j-1} + c_2 s_{j-2} + \dots + c_L s_{j-L})$$

$$C(D) = 1 + c_1 D + c_2 D^2 + \dots + c_L D^L$$

LF SR: Exercices

- 1 Montrer que la sortie d'un LFSR est ultimement périodique.
- 2 Borner la période de la séquence de sortie d'un LFSR en fonction de son nombre de registres.
- 3 Calculer les séquences binaires possibles pour un LFSR dont le polynôme de rétroaction est $P(X) = X^4 + X^2 + X + 1$. Quelles sont leurs périodes ?
- 4 Mêmes questions avec $P(X) = X^4 + X^3 + X^2 + X + 1$ et $P(X) = X^4 + X^3 + 1$.

Linear Feedback Shift Registers

Theorem

La séquence de sortie d'un LFSR à L registres est périodique avec une période inférieure à $2^L - 1$ (et ce pour tout état initial).

Polynôme primitif

Definition

Un polynôme $P(X)$ de degré $L > 0$ défini sur \mathbb{F}_2 est primitif s'il est irréductible et

$$\min \{ i > 0 \mid X^i = 1 \pmod{P} \} = 2^L - 1$$

Polynôme primitif: Exercices

Montrer qu'un polynôme primitif sur \mathbb{F}_2 :

- 1 a toujours un coefficient constant non nul ($a_0 \neq 0$),
- 2 a toujours un nombre impair de monomes,
- 3 a au moins un monome de degré impair.

La réciproque est-elle vraie?

Polynôme primitif: Exercices

Sur \mathbb{F}_2 :

- 1 Est-ce que $X^2 + X + 1$ est primitif?
- 2 Montrer que $X^6 + X^3 + 1$ n'est pas primitif.
- 3 Montrer que $X^5 + X^3 + X^2 + X + 1$ est primitif. Qu'en est-il des autres polynômes de degré 5?

LF SR de période maximale

Theorem

Un LF SR à L registres a une période $2^L - 1$ (la période maximale) ssi son polynôme de rétroaction est primitif et que l'état initial est non-nul.

LFSR de période maximale

Theorem

Un LFSR à L registres a une période $2^L - 1$ (la période maximale) ssi son polynôme de rétroaction est primitif et que l'état initial est non-nul.

Exercice

Le prouver pour un LFSR «de Galois».

Série formelle associée à un LFSR

Pour un LFSR de séquence de sortie s_0, s_1, \dots on définit la série formelle

$$s(X) = \sum_{i=0}^{\infty} s_i X^i.$$

Série formelle associée à un LFSR

Theorem

La séquence de sortie $(s_i)_{i \geq 0}$ est produite par un LFSR de Fibonacci avec polynôme de rétroaction $f(X) = 1 + c_1X + \dots + c_LX^L$ si et seulement si

$$s(X) = \frac{g(X)}{f(X)}$$

où $g(X)$ est un polynôme à coefficients dans \mathbb{F}_2 avec $\deg(g) < \deg(f)$.
De plus on peut calculer $g(X)$ avec l'état initial du LFSR:

$$g(X) = \sum_{i=0}^{L-1} X^i \sum_{j=0}^i c_{i-j} s_j.$$

Série formelle associée à un LFSR

Preuve

$$\begin{aligned} s(X) \cdot f(X) &= \left(\sum_{i \geq 0} s_i X^i \right) \left(\sum_{i=0}^L c_i X^i \right) \\ &= \sum_{i=0}^{L-1} X^i \left(\sum_{j=0}^i c_{i-j} s_j \right) + \sum_{i \geq L} X^i \left(\sum_{j=0}^L c_j s_{i-j} \right) \end{aligned}$$

Série formelle associée à un LFSR

Preuve

Par définition on a

$$s_i = \sum_{j=1}^L c_j s_{i-j}$$

donc

$$s(X) \cdot f(X) = \underbrace{\sum_{i=0}^{L-1} X^i \left(\sum_{j=0}^i c_{i-j} s_j \right)}_{g(X)} + \sum_{i \geq L} X^i \left(\underbrace{s_i + s_i}_0 \right)$$

Série formelle associée à un LFSR

Theorem

Tout séquence binaire périodique peut être générée par un LFSR.

Série formelle associée à un LFSR

Preuve

Soit $s(X) = \sum_{i \geq 0} s_i X^i$ la série formelle associée à la séquence périodique $(s_i)_{i \geq 0}$ de période ω :

$$\begin{aligned} s_{i+\omega} &= s_i \\ X^\omega s(X) &= \sum_{i \geq 0} s_i X^{i+\omega} = \sum_{i \geq 0} s_{i+\omega} X^{i+\omega} \\ &= \sum_{i \geq \omega} s_i X^i = \sum_{i=0}^{\omega-1} s_i X^i + s(X) \\ \underbrace{(X^\omega + 1)}_{f'(X)} s(X) &= \underbrace{\sum_{i=0}^{\omega-1} s_i X^i}_{g'(X)} \end{aligned}$$

Série formelle associée à un LFSR

On a montré

$$s(X) = \frac{g'(X)}{f'(X)} = \frac{g(X)}{f(X)}$$

où g et f sont premiers entre eux et définissent un LFSR produisant la séquence $(s_i)_{i \geq 0}$.

Série formelle associée à un LFSR

- 1 Décrire un LFSR de Fibonacci calculant la séquence $(s_i)_{i \geq 0}$ où

$$s(X) = \frac{X^2 + X + 1}{X^3 + X + 1}.$$

- 2 Décrire le LFSR minimal calculant la séquence répétant le motif "101011", *i. e.*

$$S = 101011101011101011101011 \dots$$

Attaques contre des LFSR

- À partir d'un certain nombre de bits du flot chiffrant, l'algorithme de Berlekamp-Massey retrouve le polynôme minimal $f(X)$ d'un LFSR qui le génère.
- En supposant que le polynôme f a un degré plus petit que L , il est nécessaire de connaître $2L$ bits de la séquence pour trouver f .

L'algorithme de Berlekamp-Massey a une complexité $\mathcal{O}(L^2)$. Il est possible d'utiliser un algorithme plus simple dont la complexité est $\mathcal{O}(L^3)$.

Relations linéaires

Rappel:

$$s_j = (c_1 s_{j-1} + c_2 s_{j-2} + \dots + c_L s_{j-L})$$

Supposons $s_0, s_1, \dots, s_{2L-1}$ connus. On peut écrire le système suivant:

$$s_L = (c_1 s_{L-1} + c_2 s_{L-2} + \dots + c_L s_0)$$

$$s_{L+1} = (c_1 s_L + c_2 s_{L-1} + \dots + c_L s_1)$$

$$s_{L+2} = (c_1 s_{L+1} + c_2 s_L + \dots + c_L s_2)$$

\vdots

$$s_{2L-1} = (c_1 s_{2L-2} + c_2 s_{2L-3} + \dots + c_L s_{L-1})$$

L équations indépendantes avec L inconnues, donc on peut utiliser la méthode de Gauss.

Algorithme de Berlekamp-Massey

- Entrée: séquence de bits (s_0, s_1, \dots) générés par un LFSR
- Sortie: d et des c_0, c_1, \dots, c_d tels que

$$c_0 s_t = c_1 s_{t-1} + \dots + c_d s_{t-d}$$

pour tout $t \geq d$.

- Principe: calculer des polynômes annulateurs de la séquence.

Algorithme de Berlekamp-Massey

Algorithm Berlekamp-Massey algorithm

INPUT: a binary sequence $s^n = s_0, s_1, s_2, \dots, s_{n-1}$ of length n .

OUTPUT: the linear complexity $L(s^n)$ of s^n , $0 \leq L(s^n) \leq n$.

1. *Initialization.* $C(D) \leftarrow 1$, $L \leftarrow 0$, $m \leftarrow -1$, $B(D) \leftarrow 1$, $N \leftarrow 0$.
 2. While $(N < n)$ do the following:
 - 2.1 *Compute the next discrepancy d .* $d \leftarrow (s_N + \sum_{i=1}^L c_i s_{N-i}) \bmod 2$.
 - 2.2 If $d = 1$ then do the following:
 $T(D) \leftarrow C(D)$, $C(D) \leftarrow C(D) + B(D) \cdot D^{N-m}$.
If $L \leq N/2$ then $L \leftarrow N + 1 - L$, $m \leftarrow N$, $B(D) \leftarrow T(D)$.
 - 2.3 $N \leftarrow N + 1$.
 3. Return(L).
-

Algorithme de Berlekamp-Massey

Exemple sur la séquence $s_n = (0, 0, 1, 1, 0, 1, 1, 1, 0)$.

s_N	d	$T(D)$	$C(D)$	L	m	$B(D)$	N
—	—	—	1	0	-1	1	0
0	0	—	1	0	-1	1	1
0	0	—	1	0	-1	1	2
1	1	1	$1 + D^3$	3	2	1	3
1	1	$1 + D^3$	$1 + D + D^3$	3	2	1	4
0	1	$1 + D + D^3$	$1 + D + D^2 + D^3$	3	2	1	5
1	1	$1 + D + D^2 + D^3$	$1 + D + D^2$	3	2	1	6
1	0	$1 + D + D^2 + D^3$	$1 + D + D^2$	3	2	1	7
1	1	$1 + D + D^2$	$1 + D + D^2 + D^5$	5	7	$1 + D + D^2$	8
0	1	$1 + D + D^2 + D^5$	$1 + D^3 + D^5$	5	7	$1 + D + D^2$	9