ABSTRACT: Massey's conjectured algorithm for multi-sequence shift register synthesis is proved, and its suitability for the minimal realization of any linear system is also verified.

I. INTRODUCTION

It is well known that the SLFSR (shortest linear feedback shift register) synthesis of single-sequence is of great importance in practice [1][2]. The Berlekamp–Massey algorithm gives an efficient one [2]. The problem of synthesizing multi-sequence with LFSR has been given much concern by many scholars in information and control society. J.L. Massey gave a conjectured algorithm for the SLFSR synthesis of multi-sequence in 1972. In 1985 Fen Gueliang and K.K. Tzeng also gave another one [3]. In this paper we are going to prove Massey's conjectured algorithm, and verify that it is an universal one and is suited for the minimal realization of any linear system.

II. PROOF OF MASSEY'S CONJECTURED ALGORITHM

Let $B_i = a_{i1} \ldots a_{iM}$, $i=1, \ldots, M$, be $M$ sequences of length $N$ in the field $F$ and $S_i = (a_{i1} a_{i2} \ldots a_{iM})^t$, $S = (B_1 B_2 \ldots B_M)^t$, $S_i = S_1 \ldots S_i$. Then the Massey's conjectured algorithm in Fig. 1 can be stated as

MASSEY'S CONJECTURE: Assume that $(f_i, l_i)$ is the SLFSR which generates $S_i$, and $d_i = f_i(s_{i+1})$ is the $i$th discrepancy, $i=0, \ldots, n$. Then

(i) if $d_n = 0$, then $l_{n+1} = l_n$ and $f_{n+1} = f_n$.

(ii) if $d_n \neq 0$, and $d_n$ is a linear combination of $d_i, i=0, \ldots, n-1$, let $d_n = \sum_{i=1}^{r} u_i \alpha_1^i \alpha_2^i \ldots \alpha_r^i$, $I = \{i : u_i \neq 0, 1 \leq i \leq r\}$ and $(k_1, k_2, \ldots, k_r)$ is maximal in alphabetic order. Let

$$d_n = \sum_{i=1}^{r} u_i \alpha_1^{i} \alpha_2^{i} \ldots \alpha_r^{i}, \quad I = \{i : u_i \neq 0, 1 \leq i \leq r\}$$

then $l_{n+1} = \max \{l_n, \max \{n-k_i^+ \alpha_i^j : i \in I\}\}, f_{n+1} = f_n + \sum_{i=1}^{r} u_i x^{n-k_i} f_{k_i}$

© Springer-Verlag Berlin Heidelberg 1988
(iii) if $d_n$ is not a linear combination of $d_i$, $i=0$, ..., $n-1$, then $l_{n+1} = n+1$ and $f_{n+1}$ can be any polynomial in $F[x]$ of degree $n+1$.

First, we give some notations and simple results:

Let $f_i = l_1 x^{i-1} + \cdots + f_{i-1} x^1$, and

\[
\begin{pmatrix}
0 & \cdots & 0 & f_1 & f_2 & \cdots & f_{n+1}
\end{pmatrix}^t
\]

be a vector of length $n+1$. Denote $D_{n+1} = (d_0 \quad d_1 \quad \cdots \quad d_n)^t$, $A_{n+1} = (s_1 \quad s_2 \quad \cdots \quad s_{n+1})^t$ and $F_{n+1} = (f_0 \quad f_1 \quad \cdots \quad f_n)^t$. Then it is easy to know that

(i) $F_{n+1}$ is a lower triangular matrix, and is invertible.

(ii) $D_{n+1} = F_{n+1} A_{n+1}$, $A_{n+1} = C_{n+1} D_{n+1}$

where $C_{n+1} = F^{-1}$, and is also a lower triangular matrix.

Let us split the matrices $F_{n+1}, G_{n+1}, D_{n+1}$ and partition them by writing

\[
F_{n+1} = 
\begin{pmatrix}
F_n & 0 \\
G_n & 1
\end{pmatrix}, \quad G_{n+1} = 
\begin{pmatrix}
G_n & 0 \\
g_n & 1
\end{pmatrix}, \quad D_{n+1} = 
\begin{pmatrix}
D_n \\
d_n
\end{pmatrix}
\]

Define matrix $U_{(n-L+1) \times (n+1)}$ as

\[
\begin{bmatrix}
0 & 0 & \cdots & 0 & u_L & \cdots & u_1 & 1 \\
0 & 0 & \cdots & u_L & \cdots & u_1 & 0 & \cdots & 0 \\
\vdots & & & & & & \vdots \\
0 & u_L & \cdots & u_1 & 0 & \cdots & 0 \\
u_L & \cdots & u_1 & 0 & \cdots & 0
\end{bmatrix} =
\begin{bmatrix}
B & 1 \\
U_{(n-L) \times n} & 0
\end{bmatrix}
\]

where $B = (0 \quad 0 \quad \cdots \quad 0)^t$, $O = (0 \quad 0)^t$. By definition, it is apparent that

the following theorem 1 holds.

**Theorem 1.** Let $f(x) = 1 + u_1 x + \cdots + u_L x^L (L < n+1)$, then $(f,L)$ generates $S^{n+1}$ if and only if $U_{(n-L) \times n} G_{n} D_{n} = 0$ and $B G_{n} D_{n} + g_{n} D_{n} + d_{n} = 0$.

**Theorem 2.** If $(f,L)$ can generate $S^{n+1}$, $L < n+1$, then there must exist a vector $u$ such that

\[
f = f_n + \sum_{i=0}^{n-1} u_i x^{n-i} f_i, \quad d_n = -u D_n = -\sum_{i=0}^{n-1} u_i d_i
\]

**Theorem 3.** Assume that $(f_i,L)$ is the SLFSR which generates $S^i$, $i=0$, ..., $n$. Then $l_{n+1} = n+1$ if and only if $d_n$ is not a linear combination of $d_i$, $i=0$, ..., $n-1$.

**Theorem 4.** Assume that

\[
g = f_n + \sum_{i=1}^{s} u_i x^{n-k_i} f_{k_i}, \quad u_i \neq 0, \quad i=1, \ldots, s.
\]

Let $l'_i$ be the shortest $L$ such that $(f_i,L')$ can generate $S^i$. If $(g,L)$ generates
Sn+1, then we have

\[ L \geq \max \{ l'_m, n-k_1+l'_k, \ldots, n-k_s+l'_k \} = \max \{ l_n, n-k_1+l_k, \ldots, n-k_s+l_k \} \]

In order to prove theorem 4, we now prove the following lemma:

Lemma: Assume \( G = f_m + u_1 x^{m-k} f_k, u_1 \neq 0, k_1 < m, \) and \((f_m, l_m), (f_k, l_k)\) are the SLFSR's which generate \( S^m \) and \( S^k \) respectively, then if \((g, L)\) generate \( S^{m+1}\), we have

\[ L \geq \max \{ l'_m, m-k_1+l'_k \} = \max \{ l'_m, m-k_1+l_k \} \]

Proof: From the definition of \( l' \) and \( l \), we obtain that \( l'_m = l_m \) and \( l'_k = l_k \).

Because \( L \geq l_m \), so \( L \geq l'_m \). Suppose \( l'_m \leq L < m-k_1+l'_k \). Let \( j \) be the last \( j \) such that \( f_k, j \neq 0 \).

1) if \( j+m-k_1 \leq l'_m \), because \( L \geq l'_m \), so \( L-m-k_1 \geq l'_m-m+k_1 \geq j \). Put \( LL=L-m+k_1 \) and

\[ h(x) = h_1 x + \ldots + h_j x^j, \] where \( h_i = f_{k,i}, i=1, \ldots, j. \) Then

\[ g(x) = f_m + u_1 x^{m-k} h(x), j \leq LL < l'_k. \]

Because \((g, L)\) generates \( S^m \) and \( L \geq l_m \), so \( g(S^m) = \ldots = g(S^{l+1}) = 0, \) and \( f(S^m) = \ldots = f(S^{l+1}) = 0. \) Thus \( h(S^k) = \ldots = h(S_{LL+1}) = 0. \) This means that \((h, LL) = (g, LL)\) generate \( S^k \), but \( LL < l'_k. \) It is contrary to the minimality of \( l'_k \), hence \( L \geq m-k_1+l'_k = \max \{ l'_m, m-k_1+l'_k \} = \max \{ l'_m, m-k_1+l_k \}. \)

2) if \( j+m-k_1 > l'_m \), regard \( g(x) \) and \( f_m + x^{m-k} f_k \) as polynomials of degree \( m, \) \( m \leq n. \) Then the degenerating terms of \( f_m + x^{m-k} f_k \) is \( m-(m-k_1+j) \), so \( m-L \leq m-(m-k_1+j) \). Put \( LL=L-m+k_1 \), then \( j \leq LL < l'_k. \) For the same reason we know that \((h(x), LL)\) generates \( S^k \), but \( LL < l'_k. \) This is also contrary to the minimality of \( l'_k \).

Thus \( L \geq \max \{ l'_m, m-k_1+l'_k \} = \max \{ l'_m, m-k_1+l_k \}. \)

PROOF OF THEOREM 4: By using the above lemma and induction on \( s \), it is not difficult to see that Theorem 4 is true.

Theorem 5. Let \( (f_i, l_i) \) be the SLFSR which generate \( S^i \), and \( d_i = f_i(S^{i+1}), i=0, \ldots, n-1, n. \) If \( d_i(\neq 0) \) can be expressed as a linear combination of \( d_i, i=0, \ldots, n-1, \) say \( d_n = \sum u_i d_i \). Let \( I_u = \{ 0 \leq i \leq n-1 : u_i \neq 0 \}. \) Put

\[ l_{n+1} = \min_u \max \{ l_n, n-i+1 ; i \in I_u \} \]

\( \cdots (\&) \)
\[
  f_{n+1} = f_n + \sum_{i=0}^{n-1} u'_i x^{n-i} f_i
\]

where \( u' = (u'_0, \ldots, u'_{n-1}) \) is the vector which makes the right side of (\&) take its minimal value. Then \((f_{n+1}, l_{n+1})\) is a shortest LFSR that generates \( S^{n+1} \).

**Proof:** Let \( L \) denote the right side of (\&). It is obvious that \( l_{n+1} \leq L \). Let \((f_{n+1}, l_{n+1})\) be a SLFSR that generates \( S^{n+1} \), and \( l_{n+1} = n+1 \). By Theorem 2 there must exist a vector \( u \) such that

\[
  f = f_n + \sum_{i=0}^{n-1} u_i x^{n-i} f_i, \quad d_i = -uD_n = -\sum_{i=0}^{n-1} u_i d_i.
\]

Then Theorem 4 tells us that \( l_{n+1} \geq \max \{l_n, n-k_i+l_{k_i} : i \in I_u \} \geq L \), therefore \( l_{n+1} = L \). Thus \((f_{n+1}, l_{n+1})\) is a SLFSR which generates \( S \).

From the base chosen in Massey's algorithm and Theorem 5 we can easily conclude that the part (i) in Massey's algorithm is true. Part (iii) has been proved in Theorem 3. Part (i) is apparently true. Thus we have completely proved Massey's conjectured algorithm until now.

Let \( V \) be a vector space over the field \( F \), \( S = s_1 \ldots s_n \) be a vector sequence of length \( n \). The problem of finding a pair \((f_n(x), l_n)\) such that \((f_n, l_n)\) generates \( S \) and \( l_n \) is minimal is referred to as the problem of minimal realization for vector sequence.

Notice that the proofs of all the theorems and lemma is independent of what the \( s_i \)'s are, but only require that \( s_i \)'s belong to a vector space over \( F \). So all the results are true for vector sequence. This means that Massey's algorithm is an universal one, it is suited for the minimal realization of any linear system. We now give some special cases of the universal algorithm:

1) If \( V = F \), then it is the B-M algorithm.

2) If \( V = F^m \), then it is the Massey's one for multi-sequence LFSR synthesis.

3) If \( V = F_{nxn} \), then it gives a minimal realization algorithm for matrix sequence.

4) If \( F = GF(q) \), \( V = GF(q^m) \), then it gives a minimal realization algorithm for the sequence in \( GF(q^m) \) over \( GF(q) \).

**ACKNOWLEDGMENT**

The author wishes to thank Prof. Xiao Guozhen for his guidance. Also much thanks to Shan Weijuan, Guo Baolan and the people in the 'Seminar for the Theory of Coding and Cryptology' for their helpful suggestions.
REFERENCES


Fig. 1 Massey's Conjectured Algorithm for Multi-sequence Shift Synthesis.