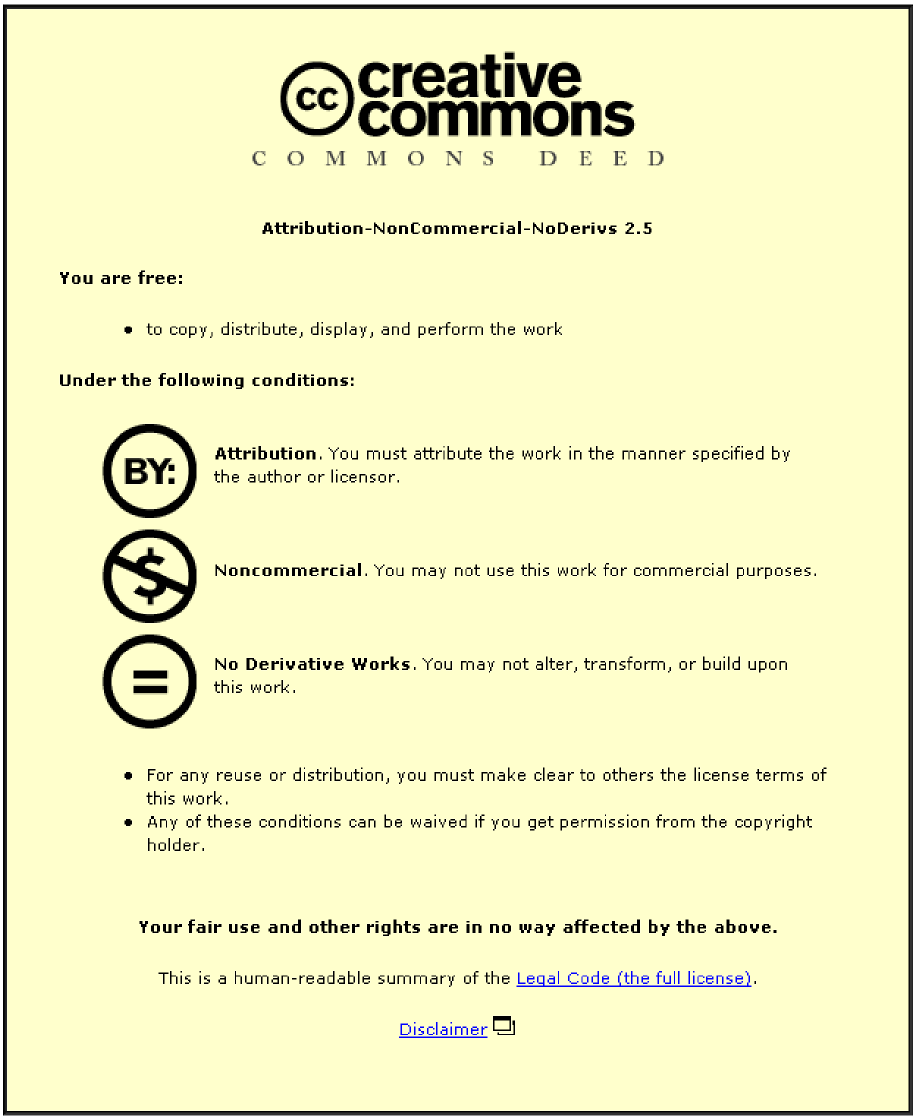

# Linear Complexity for Sequences with Characteristic Polynomial $f^v$

Alex J. Burrage, Ana Sălăgean, Raphael C.-W. Phan

*Abstract*—**We present several generalisations of the Games-Chan algorithm. For a fixed monic irreducible polynomial $f$ we consider the sequences $s$ that have as characteristic polynomial a power of $f$. We propose an algorithm for computing the linear complexity of $s$ given a full (not necessarily minimal) period of $s$. We give versions of the algorithm for fields of characteristic 2 and for arbitrary finite characteristic $p$, the latter generalising an algorithm of Kaida *et al.* We also propose an algorithm which computes the linear complexity given only a finite portion of $s$ (of length greater than or equal to the linear complexity), generalising an algorithm of Meidl. All our algorithms have linear computational complexity. The algorithms for computing the linear complexity when a full period is known can be further generalised to sequences for which it is known a priori that the irreducible factors of the minimal polynomial belong to a given small set of polynomials.**

## I. Introduction

The linear complexity of a sequence and the minimum linear recurrence relation (equivalently minimum LFSR) are important parameters for many applications, including cryptography.

The well-known Berlekamp-Massey algorithm computes the linear complexity of a sequence in quadratic time. For certain classes of sequences more efficient algorithms exist. The Games-Chan algorithm [1] takes linear time and works for binary sequences with period of the form $2^n$. It exploits the fact that in this case the minimal polynomial is a factor of $x^{2^n} - 1 = (x-1)^{2^n}$ hence it is a power of $x-1$ and we only need to determine which power. The Games-Chan algorithm assumes that we know a whole (not necessarily minimal) period of the sequence. In Algorithm 1 we generalise the Games-Chan algorithm to the case when it is known a priori that the minimal polynomial is a power of a certain fixed irreducible polynomial $f$ (so the Games-Chan algorithm would be the case $f = x-1$).

The Games-Chan algorithm has been generalised to fields of arbitrary characteristic by Kaida *et al.* in [2] and we will similarly give Algorithm 2 which is the generalisation of Algorithm 1 to arbitrary characteristic. (Our algorithm reduces to the one of Kaida *et al.* in [2] when $f = x-1$.)

It was noted by Sălăgean in [3] and by Meidl in [4] that we actually do not need to have a whole period of the sequence in order to determine its linear complexity using the Games-Chan algorithm. It suffices to have a number of terms greater or equal to the linear complexity, provided we still know that the sequence admits as characteristic polynomial a power of $x-1$ or more generally of some irreducible polynomial $f$. For finite sequences which have a characteristic polynomial of the form $f^v$ Meidl gives two algorithms in [4]: one for $f = x-1$ and arbitrary $v$, the other for arbitrary $f$ and $v$ being a power of 2. We generalise his approach as Algorithm 3, which works for arbitrary $f$ and arbitrary $v$. At first sight it would seem tempting to take this generalisation further, to $k$-error linear complexity, as in [4, Section 4]. However, we do not feel that such work would be worthwhile, as the definition used for the k-error complexity in [4] is a restricted one, (it computes the minimum linear complexity of all sequences $z$ at Hamming distance $k$ from $s$ with the additional condition that $z$ admits as characteristic polynomial a power of $f$) and is not equivalent to the generally used definition (except for $f = x-1$).

We further generalise our algorithms for infinite sequences to determine the minimal polynomial when all its irreducible factors are known a priori (Algorithm 4). This algorithm is efficient only if the number of irreducible factors is small.

For all the algorithms and their proofs we found it convenient to use the action of a polynomial on a sequence (Definition 3), a notion that has been used in different guises in several papers. We felt the proofs were shorter and simpler this way compared to the original proofs of many of the algorithms we generalised.

## II. Preliminaries

The linear complexity of a sequence is defined as usual:

**Definition 1.** *Given an infinite sequence $s = s_0, s_1, \ldots$ (or a finite sequence $s = s_0, s_1, \ldots, s_{m-1}$ ) with elements in a field $K$ we say that $s$ is a* linear recurrent sequence *if it satisfies a* homogeneous linear recurrence relation*, i.e. there are $c_0, c_1, \ldots, c_{L-1} \in K$ such that*

$$s_j + c_{L-1}s_{j-1} + \ldots + c_1 s_{j-L+1} + c_0 s_{j-L} = 0$$

*for all $j = L, L+1, \ldots$ (or for all $j = L, L+1, \ldots m-1$, respectively). We associate to it a* characteristic polynomial *$C(x) = x^L + c_{L-1}x^{L-1} + \ldots + c_1 x + c_0$. If $L$ is minimal for the given sequence, we call $L$ the* linear complexity *of $s$ and we call $C(x)$ a* minimal polynomial*.*

For infinite sequences the minimal polynomial is unique and any other characteristic polynomial is a multiple of the minimal polynomial.

Throughout this paper we work in a field $K$ of finite characteristic $p$. We denote by $s = s_0, s_1, \ldots$ an infinite sequence over $K$ and by $s'$ a finite sequence consisting of

Alex J. Burrage and Ana Salagean: Computer Science, Loughborough University, Leicestershire, LE11 3TU, UK. Raphael C.-W. Phan: Electronic and Electrical Engineering, Loughborough University, Loughborough LE11 3TU, UK (email: a.burrage@lboro.ac.uk, a.m.salagean@lboro.ac.uk, r.phan@lboro.ac.uk)

successive terms of $s$. The infinite sequence $s$ will be assumed to be periodic with period $N$. The finite or infinite sequence consisting only of zeroes will be denoted by $\mathbf{0}$ (the length of the sequence will be clear from the context). A monic irreducible polynomial $f \in K[x]$, different from 1 and $x$ is fixed throughout. For a polynomial $g$ we denote by $\mathrm{wt}(g)$ the weight of $g$, i.e. the number of non-zero coefficients of $g$ (by analogy with the Hamming weight of vectors).

For convenience we will introduce the following notation:

**Definition 2.** *Let $g$ be a monic polynomial in $K[x]$. We define $\mathcal{M}(g)$ to be the set of all infinite sequences over $K$ with characteristic polynomial equal to $g$. We also define $\mathcal{M}(g^\infty) = \cup_{i=0}^{\infty}\mathcal{M}(g^i)$.*

The following definition is a commonly used notion:

**Definition 3.** *Let $g = \sum_{i=0}^{n} a_i x^i$ be a polynomial.*

*For an infinite sequence $s$ we define the action of $g$ on $s$, denoted $gs$, to be the infinite sequence $t = t_0, t_1, \ldots$ defined by $t_i = \sum_{j=0}^{n} a_j s_{i+j}$.*

*For a finite sequence $s' = (s_0, s_1, \ldots, s_{m-1})$ with $m > n$ we define the action of $g$ on $s'$, denoted $gs'$, to be the finite sequence $t' = (t_0, t_1, \ldots, t_{m-n-1})$ defined by $t_i = \sum_{j=0}^{n} a_j s_{i+j}$. (One could extend the definition to $m \leq n$ but this situation will not occur in this paper).*

Using the terminology of actions, the following results concerning characteristic polynomials are immediate:

**Lemma 1.** *Let $g \in K[x]$ be monic and let $s$ be an infinite sequence. Then:*
*(i) $g$ is a characteristic polynomial of $s$ iff $gs = \mathbf{0}$. Moreover, $g$ is the minimal polynomial of $s$ iff $g$ is a polynomial of minimal degree for which $gs = \mathbf{0}$.*
*(ii) Let $h \in K[x]$. Let $\gcd(g, h) = g_2$ and $g = g_1 g_2$ with all polynomials monic. If $g$ is the minimal polynomial of $s$ then $g_1$ is the minimal polynomial of $hs$.*
*(iii) If $s$ is periodic and $N$ is a period of $s$ then $N$ is also a period of $gs$.*

*Proof:* Parts (i) and (iii) are clear. For (ii) write $h$ as $g_2 h'$. Denote the minimal polynomial of $hs$ by $g_3$. We will prove that $g_1 = g_3$. From (i) we know $gs = g_1 g_2 s = \mathbf{0}$ and $g_3 hs = g_3 g_2 h' s = \mathbf{0}$. Since the minimal polynomial of a sequence divides any other characteristic polynomial, $g_1 g_2 | g_3 g_2 h'$. Since $h'$ and $g_1$ are coprime we have $g_1 | g_3$. On the other hand, $gs = \mathbf{0}$ implies $gh's = g_1 g_2 h' s = g_1 hs = \mathbf{0}$ and since $g_3$ is the minimal polynomial of $hs$, we have $g_3 | g_1$, which combined with $g_1 | g_3$ proved previously results in $g_3 = g_1$. ■

Based on the well-known exponentiation rule (sometimes known as "Freshman's dream"): $(a + b)^p = a^p + b^p$ for all $a, b \in K$, we have:

**Lemma 2.** *Let $g$ be a polynomial over a field $K$ of finite characteristic $p$. Then $\mathrm{wt}(g^i) \geq \mathrm{wt}(g)$ with equality if $i$ is a power of $p$.*

Recall that the order of a polynomial $g$ (with $x \nmid g$), denoted $\mathrm{ord}(g)$ is the smallest integer $m$ such that $g$ is a factor of $x^m - 1$. The minimal period of a periodic sequence is the same as the order of its minimal polynomial. The order of an irreducible polynomial $f$ is $p^{\deg(f)} - 1$ or a factor thereof; hence $\mathrm{ord}(f)$ is not divisible by $p$. The order of a power of an irreducible polynomial can be derived as follows:

**Theorem 1.** *([5, Theorem 3.8]) Let $f$ be irreducible over $K[x]$ and let $r$ be a positive integer. Then $\mathrm{ord}(f^r) = p^t \,\mathrm{ord}(f)$ where $t$ is the smallest integer with $p^t \geq r$.*

## III. Linear Complexity of Infinite Sequences with Characteristic Polynomial a Power of an Irreducible Polynomial

We assume that we are given an infinite sequence $s$ of (not necessarily minimal) period $N$ and that we know that the sequence admits as characteristic polynomial a power of a fixed irreducible polynomial $f$, i.e. $s \in \mathcal{M}(f^\infty)$. Our goal is to determine the minimal polynomial of $s$, which will obviously be of the form $f^r$ for some integer $r$. A naive algorithm could compute $f^i s$ for increasing values of $i$ (by repeatedly replacing $s$ by $fs$) until the zero sequence is obtained. The more efficient method described here finds an upper bound for $r$ and then does a $p$-ary search for the value of $r$.

An upper bound on the value of $r$ can be obtained by looking at the period $N$ and using Theorem 1:

**Lemma 3.** *Let $s \in \mathcal{M}(f^\infty)$ and let $N$ be a (not necessarily minimal) period of $s$. Write $N$ as $N = p^w N'$ with $p \nmid N'$. Then $s \in \mathcal{M}(f^{p^w})$, $\mathrm{ord}(f) | N'$ and $p^w \,\mathrm{ord}(f)$ is also a period of $s$.*

*Proof:* Let $f^r$ be the minimal polynomial of $s$ and let $w'$ be the smallest integer with $p^{w'} \geq r$. The minimal period of $s$ is $\mathrm{ord}(f^r)$, which by Theorem 1 equals $p^{w'}\,\mathrm{ord}(f)$. Any other period of $s$, for example $N = p^w N'$ is a multiple thereof. Since neither $\mathrm{ord}(f)$ nor $N'$ are divisible by $p$, this means $w' \leq w$ and $\mathrm{ord}(f) | N'$. Hence $r \leq p^{w'} \leq p^w$ so $s \in \mathcal{M}(f^{p^w})$ and $p^w \,\mathrm{ord}(f)$ is a period of $s$. ■

Once we have an upper bound for $r$ we can find the exact value of $r$ by a $p$-ary search. We actually find the largest exponent $i$ for which $f^i s \neq \mathbf{0}$, i.e. $r - 1$. To obtain $r$, a correction of $+1$ is added at the end. One can view the $p$-ary search equivalently as determining the digits of the base $p$ representation of $r - 1$. When testing whether $f^i s \neq \mathbf{0}$ for different values of $i$, the values of $i$ which are powers of $p$ are preferred for efficiency reasons, as they minimise the weight of $f^i$, see Lemma 2.

**Lemma 4.** *Let $s \in \mathcal{M}(f^\infty)$, $s \neq \mathbf{0}$ and let $N = p^w N'$ with $p \nmid N'$ be a (not necessarily minimal) period of $s$ (which by Lemma 3 means $s \in \mathcal{M}(f^{p^w})$). Let $f^r$ be the minimal polynomial of $s$ and let $r - 1 = r_{w-1}p^{w-1} + r_{w-2}p^{w-2} + \ldots + r_1 p + r_0$ with $r_i \in \{0, 1, \ldots, p-1\}$ be the representation of $r - 1$ in base $p$. If $w = 0$ then $r = 1$. For $w \geq 1$ we have: $r_{w-1}$ is the largest integer $i \geq 0$ for which $f^{ip^{w-1}} s \neq \mathbf{0}$. Moreover, putting $t = f^{r_{w-1}p^{w-1}} s$ we have $t \in \mathcal{M}(f^{p^{w-1}})$, $t$ has minimal polynomial $f^{r - r_{w-1}p^{w-1}}$ and period $N/p$.*

*Proof:* Write $N = p^w N'$ with $p \nmid N'$. By Lemma 3, we know $\mathrm{ord}(f) | N'$. It is easy to see that $r_{w-1}p^{w-1} \leq r - 1 < (r_{w-1} + 1)p^{w-1}$, hence $r_{w-1}p^{w-1} < r \leq (r_{w-1} + 1)p^{w-1}$. So $f^{r_{w-1}p^{w-1}}$ is not a characteristic polynomial of $s$, whereas

$f^{(r_{w-1}+1)p^{w-1}}$ is. By Lemma 1(i), this means $f^{r_{w-1}p^{w-1}}s \neq \mathbf{0}$ and $f^{(r_{w-1}+1)p^{w-1}}s = \mathbf{0}$.

The last equality also means that $f^{p^{w-1}}t = \mathbf{0}$, i.e. $t \in \mathcal{M}(f^{p^{w-1}})$. By Theorem 1, $p^{w-1}\operatorname{ord}(f)$ is a period of $t$ and since $\operatorname{ord}(f)|N'$, $p^{w-1}N' = N/p$ is also a period of $t$. ■

From Lemmas 3 and 4 we have:

**Corollary 1.** *With the notations of Lemma 4: $r_{w-1} = 0$ iff $f^{p^{w-1}}s = \mathbf{0}$ iff $s \in \mathcal{M}(f^{p^{w-1}})$ iff $s$ has period $N/p$.*

Based on the Lemmas and Corollary above we present the following two algorithms LinCompChar2 and LinComp given as Algorithms 1 and 2. The first is for $p = 2$ and the second for arbitrary $p$ (including $p = 2$).

Note that throughout the algorithms the current value of the infinite sequence $s$ is implicitly stored as being the finite sequence $s' = (s_0, s_2, \ldots, s_{N-1})$ of length $N$ repeated periodically. When computing the action of a polynomial say $g = \sum_{i=0}^{n} a_i x^i$ on the infinite $s$ thus stored, the result will be an infinite sequence $t$ of period $N$ stored as the finite sequence $t' = (t_0, t_2, \ldots, t_{N-1})$ consisting of the first $N$ terms of $t$, computed from $s'$ as $t_i = \sum_{j=0}^{n} a_j s_{(i+j) \bmod N}$.

**Algorithm 1** LinCompChar2($s', N, f$)

```
Input: $f \in K[x]$ an irreducible polynomial over a field $K$ of characteristic 2; $s' = (s_0, \ldots, s_{N-1})$ a finite sequence over $K$ consisting of the first $N$ terms of an infinite sequence $s$ of (not necessarily minimal) period $N$ such that $s \in \mathcal{M}(f^\infty)$.
Output: The minimal polynomial of the sequence $s$
begin
$C = 0$
if $s' = \mathbf{0}$ then
    return($f^C$)
end if
$w =$ the largest integer for which $2^w$ divides $N$
Optionally, if $\operatorname{ord}(f)$ precomputed, set $N = 2^w \operatorname{ord}(f)$
while $w \geq 1$ do
    $t' = f^{2^{w-1}} s'$ (as action on an infinite sequence)
    if $t' \neq \mathbf{0}$ then
        $s' = t'$
        $C = C + 2^{w-1}$
    end if
    $s' = (s_0, s_1, \ldots, s_{N/2-1})$
    $w = w - 1$
    $N = N/2$
end while
$C = C + 1$
return($f^C$)
end
```

**Theorem 2.** *Algorithms LinCompChar2 and LinComp (Algorithms 1 and 2) are correct and terminate. Their complexity is $\mathcal{O}(N)$ if we consider $f$ to be fixed, or $\mathcal{O}(\mathrm{wt}(f)N)$ if $f$ is an input parameter.*

**Algorithm 2** LinComp($s', N, f$)

```
Input: $f \in K[x]$ an irreducible polynomial over a field $K$ of characteristic $p$; $s' = (s_0, \ldots, s_{N-1})$ a finite sequence over $K$ consisting of the first $N$ terms of an infinite sequence $s$ of (not necessarily minimal) period $N$ such that $s \in \mathcal{M}(f^\infty)$.
Output: The minimal polynomial of the sequence $s$
begin
$C = 0$
if $s' = \mathbf{0}$ then
    return($f^C$)
end if
$w =$ the largest integer for which $p^w$ divides $N$
Optionally, if $\operatorname{ord}(f)$ precomputed, set $N = p^w \operatorname{ord}(f)$
while $w \geq 1$ do
    $t' = f^{p^{w-1}} s'$ (as action on an infinite sequence)
    while $t' \neq \mathbf{0}$ do
        $s' = t'$
        $C = C + p^{w-1}$
        $t' = f^{p^{w-1}} s'$ (as action on an infinite sequence)
    end while
    $s' = (s_0, s_1, \ldots, s_{N/p-1})$
    $w = w - 1$
    $N = N/p$
end while
$C = C + 1$
return($f^C$)
end
```

*Proof:* The termination follows from the fact that the value of $w$ is decreased by one at each run of the while loop. Additionally, for characteristic $p$, the inner while loop will run at most $p - 1$ times, as we know that before it starts we have $f^{p^w}s = 0$. The correctness follows from Lemmas 3, 4 and Corollary 1.

For the complexity it suffices to show the second result. Let $N_0$ be the initial value of $N$, $w_0$ be the initial value computed for $w$ and let $N' = N_0/p^{w_0}$. The while loop will run $w_0$ times.

In the binary case, the complexity of each individual loop is dominated by the calculation of $t' = f^{2^{w-1}}s'$, a finite sequence representing the first $2^w N'$ terms of an infinite sequence of period $2^w N'$. The number of summands for each term is fixed by Lemma 2 as $\mathrm{wt}(f)$. So the number of arithmetic operations is $2^w N'(\mathrm{wt}(f) - 1)$. For characteristic $p$ each loop takes $(\mathrm{wt}(f) - 1)p^{w+1}N'$ steps in the worst case.

In total we have $\sum_{w=1}^{w_0} 2^w N'(\mathrm{wt}(f) - 1) = 2(2^{w_0} - 1)N'(\mathrm{wt}(f) - 1) \leq 2N_0\,\mathrm{wt}(f)$ for the binary case and $(\mathrm{wt}(f) - 1)N' \sum_{w=1}^{w_0} p^{w+1} = (\mathrm{wt}(f) - 1)p^2 N' \frac{p^{w_0-1}}{p-1}$ for arbitrary $p$. ■

Alternative algorithms can be obtained for LinCompChar2 and LinComp (Algorithms 1 and 2) by using the last equivalence of Corollary 1. Namely, we can check immediately at the start of the outer while loop whether the current value of $s'$ consists of $p$ repeating copies of the same sequence. If this is the case we do not compute $t'$ but skip to the instructions for updating the values of $s', w$ and $N$ at the end of the loop. The algorithms thus modified would have the same worst-case complexity but will behave slightly better for the case when $r - 1$ has many 0's in its representation in base $p$.

**Remark 1.** *For $f = x-1$, Algorithm 1 reduces to the Games-Chan algorithm, [1], as computing $t' = (x-1)^{2^{w-1}}s' = (x^{2^{w-1}}-1)s'$ for a sequence of period $2^w$ means $t'$ is the component-wise subtraction of the two halves of $s'$ (i.e. $t'$ is $L(s) - R(s)$ in the notation used in the Games-Chan algorithm). Similarly Algorithm 2 reduces to the algorithm of Kaida at al. [2].*

*Note that in the Games-Chan algorithm the final instruction $C = C+1$ is done conditionally, only if $s' \neq \mathbf{0}$. If one deals at the start of the algorithm with the case of an all-zero input sequence (as we do), it is no longer necessary to check at the end if $s' \neq \mathbf{0}$, as this will always be the case. Namely, if $s'$ is non-zero at the start of the outer while loop it will stay non-zero throughout, as each new value of $s'$ is always set to either the first $N/p$ elements of $s'$ (and $s'$ consists in this case of $p$ repeating identical sequences, see Corollary 1, which are therefore non-zero) or to a non-zero value of $t'$.*

**Example 1.** *Let $K = \mathrm{GF}(2)$ and $f = x^3 + x + 1$. The sequence $s$ has period $N = 28$ and its first 28 terms are $s' =$ 0101000 0101101 0110111 1101110. The running of Algorithm 1 is described in the table below:*

| $s'$ | $w$ | $t = \mathbf{0}$? | $C$ |
|---|---|---|---|
| *0101000 0101101*<br>*0110111 1101110* | *2* | *No* | *2* |
| *0011111 1000011* | *1* | *No* | *3* |
| *1011100* | *0* | | |
| $C = C+1$ | | | *4* |
| *Return* $f^4$ | | | |

## IV. Linear Complexity of Finite Sequences with Characteristic Polynomial a Power of an Irreducible Polynomial

It was noticed by Sălăgean in [3] and by Meidl in [4] that we actually do not need to have the whole period of the infinite sequence in the Games-Chan algorithm in order to compute the linear complexity. In this section we generalise the idea of Meidl, [4, Sections 2 and 3].

For a fixed polynomial $g$, an individual infinite sequence $s$ with characteristic polynomial $g$ is uniquely defined (within the class of all sequences with characteristic polynomial $g$) by its initial $\deg(g)$ terms. Can we decide if $s$ admits a characteristic polynomial of lower degree just by examining these initial $\deg(g)$ terms?

**Lemma 5.** *Let $s$ be an infinite sequence with characteristic polynomial $g = g_1g_2$. Then:*
*$s$ has characteristic polynomial $g_1$ iff $s' = (s_0, \ldots, s_{\deg(g)-1})$ has characteristic polynomial $g_1$.*

*Proof:* The direct implication is obvious. Conversely, assume $s'$ has characteristic polynomial $g_1$ i.e. $g_1s' = (0, \ldots, 0)$, a finite sequence of $\deg(g) - \deg(g_1) = \deg(g_2)$ terms. Note this sequence also coincides with the first $\deg(g_2)$ terms of the infinite sequence $g_1s$. By Lemma 1(i), $gs = g_2g_1s = \mathbf{0}$, so $g_1s$ has characteristic polynomial $g_2$. But then $g_1s = \mathbf{0}$ as its first $\deg(g_2)$ terms are all zero and its linear complexity is at most $\deg(g_2)$. ■

Consequently, if we are given $s'$ as being the first $v\deg(f)$ terms of a sequence $s \in \mathcal{M}(f^v)$ we can check whether $s$ admits some characteristic polynomial of lower degree, i.e. $f^{v'}$ with $v' < v$ by checking whether $f^{v'}s' = \mathbf{0}$.

The algorithm LinCompChar2Finite is given as Algorithm 3 and is similar to the Algorithm 1 in the previous section. Note that throughout the algorithm, the length of the current value of $s'$ is $v\deg(f)$ for the current value of $v$.

As in Theorem 2, we can prove that the computational complexity will be $\mathcal{O}(v)$ for a fixed $f$, or $\mathcal{O}(v\deg(f)\,\mathrm{wt}(f))$ if $f$ is an input parameter.

**Algorithm 3** LinCompChar2Finite($s', v, f$)

```
Input: A finite sequence $s'$ consisting of the first $v\deg(f)$ elements of an infinite sequence $s \in \mathcal{M}(f^v)$ where $f \in K[x]$ is a fixed irreducible polynomial over a field $K$ of characteristic 2.
Output: The minimal polynomial of the sequence $s$
begin
$C = 0$
if $s' = \mathbf{0}$ then
    return($f^C$)
end if
$w =$ the smallest integer such that $v \leq 2^w$
while $w \geq 1$ do
    $t' = f^{2^{w-1}}s'$ (as action on a finite sequence)
    if $t' \neq \mathbf{0}$ then
        $s' = t'$
        $C = C + 2^{w-1}$
        $v = v - 2^{w-1}$
        $w =$ the smallest integer such that $v \leq 2^w$
    else
        $v = 2^{w-1}$
        $w = w - 1$
        $s' = (s_0, s_1, \ldots, s_{v\deg(f)-1})$
    end if
end while
$C = C + 1$
return($f^C$)
end
```

**Example 2.** *Let $K = GF(2)$ and $f = x^3 + x + 1$. The finite sequence $s' =$ 010100001011010110 consists of the first $6\deg(f) = 18$ terms of the sequence in Example 1. The running of Algorithm 3 is described in the table below:*

| $s'$ | $w$ | $v$ | $t = \mathbf{0}$? | $C$ |
|---|---|---|---|---|
| *010100001011010110* | *3* | *6* | *Yes* | *0* |
| *010100001011* | *2* | *4* | *No* | *2* |
| *001111* | *1* | *2* | *No* | *3* |
| *101* | *0* | *1* | | |
| $C = C+1$ | | | | *4* |
| *Return* $f^4$ | | | | |

An analogue of the algorithm LinComp to work for arbitrary finite characteristic $p$ can be similarly developed but we will not go into details here as it is straightforward.

**Remark 2.** *For $f = x - 1$ and arbitrary $v$, our Algorithm 3* *reduces to Algorithm 1 of [4]. For $f = x^2 + x + 1$ and $v$ being a power of 2, it reduces to Algorithm 2 in [4] (which, as remarked at the end on Section 3 of [4] could be generalised to arbitrary $f$ and $v$ a power of 2).*

*Let us examine the relation between the algorithms in this section (for finite sequences) and the ones in the previous section (for infinite sequences). We could easily transform one problem into the other, namely, if we have a finite sequence we can generate the whole period using the given characteristic polynomial $f^v$ (note that this process can take a number of steps exponential in the length of the finite sequence) and conversely given an infinite sequence of period $N$ we could restrict to the initial $p^w \deg(f)$ terms (with $w$ maximal such that $p^w | N$), as $f^{p^w}$ is guaranteed to be a characteristic polynomial of $f$. However, the complexity of Algorithms 1 and 2 is at least $\mathcal{O}(p^w \operatorname{ord}(f) \operatorname{wt}(f))$ whereas Algorithm 3 has complexity $\mathcal{O}(p^w \deg(f) \operatorname{wt}(f))$. Since $\deg(f) \le \operatorname{ord}(f) \le p^{\deg(f)}$ with both lower and upper bounds attained for particular values of $f$, it means that the algorithms of the previous section are potentially exponentially slower than the ones in this section and should therefore be avoided (to clarify, all are linear in the size of the input, but the size of the input can be exponentially higher if we use the full period rather than the initial $v \deg(f)$ terms). We did present them though as they are direct generalisations of the Games-Chan algorithm.*

## V. Linear Complexity of Infinite Sequences whose Minimal Polynomial is a Product of Known Irreducible Factors

With a simple adjustment to the algorithms in Section III, we can greatly increase their scope, so that they can be applied to any sequence provided each of the irreducible factors of the minimal polynomial are known.

As a consequence of Lemma 1(ii) we have:

**Corollary 2.** *Assume that the minimal polynomial of a sequence $s$ is of the form $f_1^{r_1} f_2^{r_2} \dots f_m^{r_m}$, with $f_i$ distinct irreducible polynomials. Let $x_i \ge r_i$ for $i = 2, \dots, m$. Then $f_1^{r_1}$ is a minimal polynomial of the sequence $f_2^{x_2} \dots f_m^{x_m} s$.*

Therefore, if we know each of the irreducible polynomials which divide the characteristic polynomial of a sequence and we have an upper bound on the powers of each irreducible polynomial, we can use Corollary 2 and Algorithms 1 or 2, to successively determine the powers of each irreducible polynomial in the minimal polynomial. To obtain an upper bound of the power of each irreducible polynomial, note that the minimal polynomial is a factor of $x^N - 1$ where $N$ is a period of $s$. Writing $N = p^w N'$ with $p \nmid N'$ we have $x^N - 1 = (x^{N'} - 1)^{p^w}$. Putting $\Phi_{N'} = \{f \in K[x] | f$ irreducible factor of $x^{N'} - 1\}$ all irreducible factors of the minimal polynomial are in $\Phi_{N'}$ and have multiplicity at most $p^w$. The resulting algorithm LinCompSet is presented as Algorithm 4.

**Theorem 3.** *For a sequence of period $N$, and a fixed set $\Phi$, Algorithm 4 has complexity $\mathcal{O}(n)$. For a general set of $m$ elements $\Phi = \{f_1, \dots, f_m\}$, the algorithm will have complexity $\mathcal{O}((\sum_{i=1}^m \operatorname{wt}(f_i))mN)$.*

**Algorithm 4** LinCompSet($s', N, \Phi = \{f_1, \dots, f_m\}$)

```
Input: $s'$ a finite sequence consisting of the first $N$ terms of an infinite sequence $s$ of (not necessarily minimal) period $N$; $\Phi$ a superset of the set of all irreducible factors of the minimal polynomial of $s$.
Output: The minimal polynomial of $s$
begin
$w =$ the largest integer for which $p^w$ divides $N$
$g = 1$
for $i = 1, 2, \dots, m$ do
    for $j = 1, 2, \dots, m$ do
        if $j \ne i$ then
            $t' = f_j^{p^w} s'$ (as action on infinite sequences)
        end if
    end for
    $g = g * \mathrm{LinComp}(t', N, f_i)$
end for
return($g$)
end
```

*Proof:* In each of the $m$ runs of the outer for loop, the computation of $t'$ takes $((\sum_{j=1}^m \operatorname{wt}(f_j)) - \operatorname{wt}(f_i))N$. By Theorem 2, LinComp has complexity $\mathcal{O}(\operatorname{wt}(f_i)N)$, so a total of $\mathcal{O}(N \sum_{i=1}^m \operatorname{wt}(f_i))$ for each loop. ■

Note that Algorithm 4 is therefore efficient only if $\Phi$ has a small cardinality and the total weight of its elements is small.

We could remove the condition that $\Phi$ by computing $\Phi_{N'}$ as above and using it as $\Phi$ during the algorithm. The cases of interest will be the ones where $N'$ is a small constant. In the general case all efficiency advantages of the algorithm are lost as the size of the set $\Phi_{N'}$ is in the worst case $\mathcal{O}(N)$ (see [5] Theorem 3.5 and estimates for Euler's totient function).

## VI. Conclusion

We proposed algorithms for computing the linear complexity and minimal polynomial for sequences which admit as characteristic polynomial a power of a fixed irreducible polynomial $f$. They work for any field of finite characteristic and we do not necessarily need the whole period of the sequence. For $f = x - 1$ our algorithms reduce to the algorithms of Games-Chan [1], Kaida *et al.* [2] and Meidl [4]. All our algorithms have linear computational complexity.

## References

[1] R. Games and A. Chan, "A fast algorithm for determining the complexity of a binary sequence with period $2^n$," *IEEE Trans. Information Theory*, vol. 29, pp. 144–146, 1983.

[2] T. Kaida, S. Uehara, and K. Imamura, "An algorithm for the $k$-error linear complexity of sequences over $GF(p^m)$ with period $p^n$, $p$ a prime," *Inform. Comput.*, vol. 151, pp. 134–147, 1999.

[3] A. Sălăgean, "On the computation of the linear complexity and the k-error linear complexity of binary sequences with period a power of two," *IEEE Trans on Information Theory*, vol. 51, pp. 1145–1150, 2005.

[4] W. Meidl, "How to determine linear complexity and $k$-error linear complexity in some classes of linear recurring sequences," *Cryptography and Communications*, vol. 1, pp. 117–133, 2009.

[5] R. Lidl and H. Niederreiter, *Introduction to Finite Fields and Their Applications*. Cambridge university Press, 1994.