Single-tracked shaft encoders with LFSR sequences of low combinatorial complexity

invited post ICM09 paper

***Abstract*—Maximal binary sequences generated by LFSR circuits can be used to label single-tracked absolute shaft encoders. The number of detectors required to build such encoder is called the combinatorial complexity of the sequence. For a maximal sequence, this is equal to the size to the LFSR. In order to build encoders of arbitrary resolution, this approach has been generalized to non-maximal sequences. A method to obtain sequences of this type with low combinatorial complexity is developed and evaluated in this paper. The error-correcting power of codes obtained from these sequences is also studied.**

I. Introduction

In many application areas, including angular control systems, careful selection of sensing components is key to provide the best application performance. Traditionally, angular position sensing devices are classified as incremental and absolute shaft encoders. When compared to absolute encoders, incremental encoders are simpler and lightweight. However, incremental encoders present a major drawback: they require some means of synchronization in order to obtain a reference for the axis position. On the other hand, absolute encoders provide the ability to remember the object position after any power interruption. This is one of the reasons why absolute encoders are used where a high safety stardard is required. Typical applications are for speed and position control systems in aerospace and aviation, semiconductor manufacturing, medical imaging, positioning mechanisms, machine tools and robotics, inspection equipment, telescopes and other instruments. See, for example, [1]–[3].

Absolute angular measurements are carried out by transducers that output a different n-bit code word for each of a finite number e of angular positions, yielding an angular resolution of $360/e$ degrees. This imposes the obvious restriction $e \leq 2^n$. Most commercial encoders use a Gray coding in order to reduce the number of possible scanning errors during transitions. But this approach has two drawbacks. First, that it results in a multi-tracked configuration where n detectors are arranged radially with respect to the rotation axis. This implies that encoders become larger and heavier as the number of sectors e increases, rendering them unsuitable for mass critical applications. And second, that no general method is known for building such encoders with arbitrary e; only codes where e is a power of 2 can be constructed in a systematic way.

Both of these drawbacks have been considered in the literature and many solutions have been proposed. The first drawback is usually addressed by considering single-tracked configurations where n detectors are arranged tangentially with respect to the rotation axis. The construction of such encoders requires a cyclic binary sequence of length e where all subsequences of length n are pairwise different. This is the case with sequences generated by maximal linear feedback shift registers (LFSR) of size n. Absolute encoders based on LFSR sequences of length $e = 2^n - 1$ have been proposed in several papers [4]–[7]. But these sequences still suffer from the second drawback and, furthermore, they do not exhibit the Gray property. In [8], [9] single-track Gray codes were studied, resulting in a family of single-tracked absolute encoders where detectors are equispaced instead of being contiguously arranged. The resulting encoders do satisfy the Gray property. However, these codes can not be built for arbitrary e either.

In order to overcome the second drawback, a truncation method for maximal LFSR sequences was proposed in [10]. This approach requires an extra track and some additional circuitry for enabling detection and position recovery around the truncation point. With the purpose of reducing this overhead, a general method to construct non-maximal LFSR sequences of arbitrary length was introduced in [11]. The efficiency of this method was showcased in [12]. However, examples provided there evidenced that generality came at a price: it was not possible to control the number of detectors n required by the sequences obtained. The only available bounds were $\lceil \log_2 e \rceil \leq n \leq e - 1$, and both were attained for some values of e.

This paper presents an improvement over the results in [11], [12]. In particular, a method to construct sequences that can be used to build single-tracked absolute encoders of arbitrary resolution with a quasi-optimal number of encoders is developed. The main idea consists in performing a stochastic search in a space of moderate size to obtain seeds for the LFSRs constructed in [11] that are able to generate sequences with adequate properties. A combination of formal and heuristic arguments are developed to justify why a certain search space is chosen. In addition, experimental results are provided to corroborate that this method succeeds in harnessing the variability exhibited in [12].

Encoders based on LFSR sequences do not exhibit the Gray property. This means that read errors can be found during transitions between positions. If error-detecting or error-correcting capabilities could be 'embedded' into the sequence found inside the encoder, this difficulty could be overcome under the assumption that the number of errors is small when compared to the number of detectors. The feasibility of this approach is explored in this paper via experimental evaluations. In particular, the number of detectors required by LFSR sequences obtained with our method in order to span codes with certain minimum Hamming distances is studied.

Our results show that the number of detectors required is, on average, close to the theoretical lower bound given by the Hamming bound. It is worth noting that, as far as we know, this approach is original and has not been considered in the literature before.

The rest of the paper is organized as follows. Section II is devoted to preliminaries. In particular, basic results about codes, LFSRs and shaft encoders are recalled. In section III, the problem of how many detectors are required by sequences obtained with the methods from [11], [12] is analyzed and characterized in the form of an exponential gap between two sequences of means. A stochastic search algorithm capable of obtaining sequences that close this exponential gap is developed in section IV. Experimental evidence supporting the goodness of the algorithm is provided, together with a study of error-correction capabilities provided by codes obtained from these sequences. Finally, the conclusions of this work are summarized in section V. The proof of a technical lemma can be found in an appendix at the end of the paper.

Since no particular benefit is obtained by restricting ones attention to a binary field, an abstract approach will be pursued by working over an arbitrary finite field $\mathbb{F}_q$. This has the advantage of generality and makes our results useful in the hypothetical case where q-ary detectors were used to build an encoder. The reader interested in binary sequences can particularize all results by substituting $q = p = 2$ everywhere.

II. Sequences and Encoders

A. Finite sequences

Let Σ be a finite alphabet of size q. A sequence of length e over Σ is a vector $S = (s_0, \ldots, s_{e-1}) \in \Sigma^e$. The length of S will sometimes be denoted by $|S|$. We will denote by $S_{n,i}$ the subsequence $(s_i, \ldots, s_{i+n-1})$, where $0 \leq i \leq e-1$ and the indices are taken modulo e. The set $S_n = \{S_{n,i}\}_{0 \leq i \leq e-1}$ is the *code spanned by S with window n*. The smallest window n such that $|S_n| = |S| = e$ is the *combinatorial complexity of* S and is denoted by $C(S)$. For $n \geq C(S)$, S can be used to build a code by assigning to each $0 \leq i \leq e-1$ the unique code word $S_{n,i}$. It is easy to see that one has the lower bound

$$C(S) \geq \left\lceil \log_q |S| \right\rceil . \tag{1}$$

Recall that the Hamming distance between two sequences, u and v, of length n, is the number $d(u, v)$ of positions where they differ. If U is a set of sequences of length n, we denote by $d(U)$ the minimum Hamming distance between two distict elements of U. For any $k \geq 1$, the *k-th Hamming complexity* of a sequence S is the smallest n such that $|S_n| = |S|$ and $d(S_n) \geq k$. It is denoted by $H_k(S)$. If no such n exists we will set $H_k(S) = \infty$. For $n = H_k(S)$, the code S_n has minimum Hamming distance at least k and can be used to detect at least $k-1$ errors or to correct at least $\lfloor (k-1)/2 \rfloor$ errors. By definition we have that $H_1(S) = C(S)$ and if $k > k'$ then $H_k(S) > H_{k'}(S)$. The Hamming bound [13] gives the following lower bound for $H_k(S)$:

$$\frac{q^{H_k(S)}}{\sum_{j=0}^{\lfloor \frac{k-1}{2} \rfloor} \binom{H_k(S)}{j} (q-1)^j} \geq |S|. \tag{2}$$

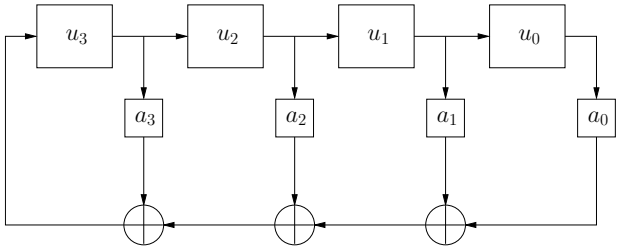

Fig. 1. An LFSR of size 4

Note that for $k = 1, 2$ this bound coincides with (1).

From now on, only sequences over a finite field $\mathbb{F}_q$ will be considered. Hence, q will be restricted to be a prime power $q = p^m$.

B. LFSR sequences

A Linear Feedback Shift Register (LFSR) is a shift register whose input bit is computed as a linear combination of its state bits (see Fig. 1). These circuits can be used to obtain infinite sequences by concatenating the value of the last tap at each successive clock cycle. Here LFSRs will be regarded as abstract machines working with elements in a finite field $\mathbb{F}_q$.

When $a_0 \neq 0$, the sequence outputted by a LFSR is periodic. The finite sequence $S = (s_0, \ldots, s_{e-1})$ correponding to the first full period of the output is the *sequence generated by the LFSR*. This sequences only depends on the feedback logic and the initial state (also called the *seed*):

$$(a_0, \ldots, a_{n-1}), (u_0, \ldots, u_{n-1}) \in \mathbb{F}_q^n.$$

Note that one has $(s_0, \ldots, s_{n-1}) = (u_0, \ldots, u_{n-1})$. The feedback logic is usually given in the form of a *connection polynomial*,

$$a(x) = x^n - (a_{n-1}x^{n-1} + \cdots + a_1 x + a_0) \in \mathbb{F}_q[X].$$

The size of the LFSR corresponds to the degree of its connection polynomial.

Given a LFSR, different seeds may generate sequences of different lengths. We will denote by $S^{a,u}$ the sequence generated by the LFSR with connection polynomial $a(x)$ and seed u. When $a(x)$ is undestood, we will just write S^u. For any fixed LFSR, the *standard seed* $v = (0, \ldots, 0, 1)$ satisfies that $|S^{a,u}| \leq |S^{a,v}|$ for any other seed u (see [11]). If equality holds we will say that u is a *maximal seed* for $a(x)$.

The *linear complexity* of a sequence S is defined as the size of the smallest LFSR that can generate S and denoted by $L(S)$. Note that $L(S) \leq |S|$ because S can be generated by the LFSR with $a(x) = x^{|S|} - 1$ and $u = S$. By construction, any sequence S^u generated by an LFSR of size n has $C(S^u) \leq n$. Therefore, in general $C(S) \leq L(S)$. Furthermore, if v is the standard seed, then $n-1 \leq C(S^v) \leq n$, since v contains $n-1$ consecutive zeros.

Note that all the complexities defined so far are invariant under a cyclic permutation of the sequence S.

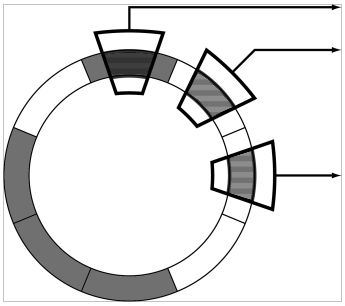

Fig. 2. A single-tracked absolute shaft encoder

C. Shaft encoders

A shaft encoder is an electro-mechanical device used to convert the angular position of a rotating rod into an electrical signal. In this paper only the family of digital single-tracked absolute encoders is considered. To build one such encoder, a ring is attached to the rod and some detectors are distributed tangentially around the ring. If an angular resolution of $360/e$ is required and the detectors used can distinguish between q different symbols, a sequence S of length e over an alphabet of size q is used to label the track in the ring. In order to identify each of the e possible positions of the ring with a unique subsequence, the number n of detectors must satisfy $n \geq C(S)$. An example with $q = 2$, $e = 8$ and $n = 3$ can be seen in Fig. 2.

These encoders are better suited for mass critical applications than the usual multi-tracked encoders based on the gray code. Due to the space saved by the reduction in the number of tracks, an important reduction of moving mass can be achieved.

To design a shaft encoder with given parameters e and q, a labeling sequence S needs to be constructed. With mass critical applications in mind, one is interested in sequences with small combinatorial complexity in order to reduce the number of detectors needed and thus, the total mass of the encoder. Although sequences with optimal complexity $C(S) = \lceil \log_q e \rceil$ are known to exist for each pair of parameters e and q [14], in general no efficient algorithm is known for finding these sequences. However, when q is a prime power and $e = q^n - 1$, maximal LFSRs can be used to generate optimal sequences with these parameters.

Extending this idea, a method based on non-maximal LFSRs was proposed in [11], [12]. This methods allows one to obtain a sequence with given parameters e and q with the smallest possible linear complexity. However, as will be shown in the following section, this method does not always deliver sequences with small combinatorial complexity.

III. The problem: an exponential gap

In [12], an algorithm to construct a sequence with given parameters e and q and smallest possible linear complexity was presented. Roughly, the algorithm used the prime factorization of e to compute a connection polynomial $a(x) \in \mathbb{F}_q[X]$ with minimal possible degree among those that can generate sequences of length e. For further reference, we will denote the procedure computing such polynomials as $\mathsf{SmallestLFSR}(e, q)$. When v is the standard seed, this polynomial satisfies that $|S^{a,v}| = e$. This was the proposed sequence. The combinatorial complexity of the sequences obtained by this method satisfies

$$\deg(a) - 1 \leq C(S^{a,v}) \leq \deg(a) = L(S^{a,v}).$$

The following lemma, whose proof can be found in the appendix, presents an easy upper bound on $C(S^{a,v})$. This, however, is exponentially away from the lower bound (1).

Lemma 1: Let $q = p^m$ and $e \geq 4$. If $a(x) = \mathsf{SmallestLFSR}(e, q)$, then $\deg(a) \leq e - 1$.

Although this upper bound is attained only for a very small fraction of lengths e, experimental evidence shows that the average combinatorial complexity of $C(S^{a,v})$ is indeed exponentially away from the theoretical lower bound $\lceil \log_q e \rceil$.

Let q be a fixed prime power and $n_e = \deg(a)$ for $e \geq 1$, where $a(x) = \mathsf{SmallestLFSR}(e, q)$. Since the degree n_e depends strongly on the multiplicative structure of e, no obvious relation exists between the dregrees obtained for different, *additively* close, e's. Hence, if one considers the sequence $(n_e)_{e \geq 1}$, it exhibits what seems a random behavior. However, the sequence of theoretical lower bounds $t_e = \lceil \log_q e \rceil$ does exhibit a regular behavior. Indeed, it is an increasing sequence. In order to be able to compare, qualitatively and quantitatively, the evolution of the combinatorial complexity with that of the lower bounds, one can consider the *sequence of means*.

Given a sequence of real numbers $(z_i)_{i \geq 1}$, its sequence of means is $(\bar{z}_i)_{i \geq 1}$, where

$$\bar{z}_i = \frac{1}{i} \sum_{1 \leq j \leq i} z_j.$$

That is, the i-th term is the mean of the first i terms from the original sequence. Usually, the mean sequence exhibits a much more regular behavior than the original sequence.

Now we consider the sequences $(\bar{n}_e)$ and $(\bar{t}_e)$. Intuitively, the terms $\bar{n}_e$ and $\bar{t}_e$ of these sequences represent the expected combinatorial complexity and the expected lower bound when one takes e' uniformly at random between 1 and e. Therefore, they can be used to evaluate the behavior of the combinatorial complexity obtained with $\mathsf{SmallestLFSR}$ for a generic e. For $q = 2$, this sequences can be seen in Fig. 4 at page 5. This plot shows that the expected combinatorial complexity grows linearly with e, $\bar{n}_e = O(e)$, while the expected lower bound grows logarithmically with e, $\bar{t}_e = O(\log e)$. This exponential gap indicates that, on average, building a shaft encoder using the sequence generated by an LFSR with connection polynomial $a(x) = \mathsf{SmallestLFSR}(e, q)$ and the standard seed v requires exponentially more detectors than the theoretical lower bound. The same behavior is found for other prime powers q. In the following section a method to overcome this exponential gap will be presented and evaluated.

IV. The solution: choosing a good seed

In what follows, a method to narrow this exponential gap will be proposed. The idea is to still use $a(x) = \mathsf{SmallestLFSR}(e, q)$ as a connection polynomial, but to choose

a seed different from the standard one. The proposed algorithm will do a search by sampling uniformly at random from a particular set of seeds. An experimental evaluation will show that, on average, the algorithm succeeds in closing the exponential gap and obtains sequences whose combinatorial complexity is within a constant factor of the optimal lower bound. Furthermore, some heuristic arguments will be developed to justify the behavior of the algorithm.

Recall from [12] that connection polynomials $a(x)$ returned by procedure SmallestLFSR are of one of these two forms:

$$a'(x) = a_1(x)\cdots a_k(x), \text{ with } k \geq 1, \quad (3)$$

$$a''(x) = a_1(x)\cdots a_k(x)a_{k+1}(x)^s, \text{ with } k \geq 0, s > 1, \quad (4)$$

where the $a_i(x)$ are monic irreducible polynomials and $a_{k+1}(x)$ has degree one. Using a formula from [15], the probability that a random seed is maximal for connection polynomials with these forms can be computed as follows:

$$p_{a'} = \frac{(q^{n_1}-1)\cdots(q^{n_k}-1)}{q^n},$$

with $n_i = \deg(a_i)$, $n = \deg(a')$, and

$$p_{a''} = \frac{(q^{n_1}-1)\cdots(q^{n_k}-1)(q^s - q^{s-1})}{q^n},$$

with $n_i = \deg(a_i)$, $n = \deg(a'')$.

The number of essentially different sequences, i.e. not cyclically equivalent, that can be generated by an LFSR with connection polynomial $a(x)$ is $p_a q^n/e$. Since, on average, n grows linearly with e, this quantity behaves asymptotically on average like q^e/e. Hence, there may be lots of essentially different sequences of length e that can be generated from $a(x)$ using different seeds, and one might expect that some of them have low combinatorial complexity. To guide the search for these sequences, seeds need to be chosen with care.

In particular, a necessary condition that a seed must satisfy in order to generate a sequence of low combinatorial complexity is to have low *non-cyclic combinatorial complexity*: a sequence $S = (s_0, \ldots, s_{e-1})$ has non-cyclic combinatorial complexity $C_N(S) = n$ if n is the smallest integer such that

$$|\{(s_i, \ldots, s_{i+n-1}) | 0 \leq i \leq e-n\}| = e - n + 1.$$

In fact, one has that $C_N(S_{n,i}) \leq C(S)$ for any subsequence $S_{n,i}$ with $n \leq |S| + C(S) - 1$, and in particular $C_N(u) \leq C(S^u)$. We say that a seed u is *good* for e if $C_N(u) \leq \lceil \log_q e \rceil$ and it is maximal for $a(x) = \mathsf{SmallestLFSR}\,(e, q)$. A seed will be called *bad* if it is not good. Generally, the standard seed v is bad since $C_N(v) = \deg(a) - 1$ which, on average, is exponentially larger than $\lceil \log_q e \rceil$. However, it is a good seed for all maximal LFSRs. Note that, in this case, all but the zero seed are good.

Since the combinatorial complexity of sequences S^u with a bad seed can never attain the optimal lower bound, it seems a good idea to search for sequences with low combinatorial complexity among those that are generated by good seeds. An easy way to obtain seeds with small non-cyclic combinatorial complexity for a general LFSR is as follows. Let $q = p^m$ be a fixed prime power and take a connection polynomial $a(x) = \mathsf{SmallestLFSR}\,(e, q)$ for some $e \geq 4$. Let $n = \deg(a)$ and $t_e = \lceil \log_q e \rceil$. If v is the standard seed and $b(x) = \mathsf{SmallestLFSR}\,(q^{t_e} - 1, q)$, then the following holds.

1: **procedure** SearchGoodSeed (e, q, δ)
2: Let $a(x) \leftarrow \mathsf{SmallestLFSR}\,(e, q)$ and $n \leftarrow \deg(a)$
3: Let $t \leftarrow \lceil \log_q e \rceil$ and $v \leftarrow (0, \ldots, 0, 1) \in \mathbb{F}_q^t$
4: Let $b(x) \leftarrow \mathsf{SmallestLFSR}\,(q^t - 1, q)$
5: Compute $S^{b,v}$
6: Compute p_a and let $M \leftarrow \lceil \log \delta / \log(1 - p_a) \rceil$
7: **for** $j \leftarrow 1 \ldots M$ **do**
8: Choose $0 \leq i \leq q^t - 2$ uniformly at random
9: Let $u \leftarrow S_{n,i}^{b,v}$ and compute $S^{a,u}$
10: **if** $|S^{a,u}| = e$ **then**
11: **return** u
12: **end if**
13: **end for**
14: **return** "Not found"
15: **end procedure**

Fig. 3. Procedure SearchGoodSeed

Proposition 2: For any $0 \leq i \leq q^{t_e} - 2$, the sequence $S_{n,i}^{b,v}$ has non-cyclic combinatorial complexity not greater than t_e.

Proof: First note that, by construction, $b(x)$ is the connection polynomial of a maximal LFSR and thus, $|S^{b,v}| = q^{t_e} - 1$ and $C(S^{b,v}) = t_e$. Now, by definition we have $t_e \geq 1$ and, by lemma 1, $q^{t_e} - 1 \geq e - 1 \geq n$. Therefore $|S^{b,v}| + C(S^{b,v}) - 1 \geq n$ and $C_N(S_{n,i}^{b,v}) \leq t_e$. ∎

This proposition provides a way to construct potentially good seeds which is exploited by the procedure found in Fig. 3. The procedure SearchGoodSeed receives as input sequence parameters e and q, and a failure probability $0 < \delta < 1$ chosen by the user. Smaller values of δ give more time to the algorithm to perform the search. This is a stochastic search that looks for good seeds of the form given by proposition 2. In words, SearchGoodSeed computes $a(x)$ and $b(x)$ and then it samples at most $\lceil \log \delta / \log(1 - p_a) \rceil$ seeds of the form $S_{n,i}^{b,v}$ while checking if any of them is good. As soon as it finds a good seed, the procedure finishes returning it. If none is found, it returns "Not found". The choice of M, the number of iterations, is made according to the following.

Assumption 3: If $u \in \mathbb{F}_q^n$ is chosen uniformly at random, the events "$S^{a,u}$ has length e" and "$u = S_{n,i}^{b,v}$ for some i" are independent.

If this assumption holds, the probability that $|S^{a,u}| = e$ when $u = S_{n,i}^{b,v}$ is then equal to p_a. In this case, the choice of M guarantees that SearchGoodSeed (e, q, δ) will return "Not found" with probability at most δ.

A. Experimental evaluation

An experimental evaluation of the procedure SearchGoodSeed was performed. A range of 1200 values was considered. For each $4 \leq e \leq 1203$, we executed SearchGoodSeed $\left(e, 2, 10^{-2}\right)$ ten times. For each of these values of e, at least one good seed was found in these ten executions. Furthermore, from the total number of calls performed, only a 0.5% of them returned "Not found". This

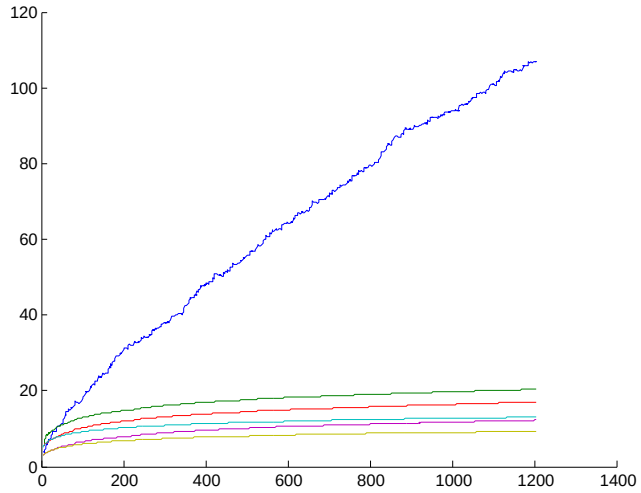

Fig. 4. From top to bottom, plots of $\bar{n}_e$, $\bar{h}_e^3$, $\bar{h}_e^2$, $\bar{b}_e^3$, $\bar{c}_e$ and $\bar{t}_e$.

shows that assumption 3 may be close to true, at least for almost all the values of e considered.

The reduction in combinatorial complexity achieved by these good seeds was evaluated by means of the sequence (c_e). For each e, the number c_e is obtained as the minimum among the combinatorial complexities of the sequences generated by the good seeds obtained for e. The mean sequence $(\bar{c}_e)$ can be seen in Fig. 4. Note that this sequence has the same log-like shape as $(\bar{t}_e)$ and, indeed, is close to this optimal sequence. In fact, the optimal value $c_e = \lceil \log_2 e \rceil$ was attained for a 21.2% of the values of e considered. Furthermore, in more than a 91% of the cases we found that $c_e \leq 1.5t_e$. These results justify our claim that SearchGoodSeed succeeds in closing the exponential gap between (n_e) and (t_e) by obtaining sequences whose combinatorial complexity is, on average, within a constant factor of the optimal one.

In order to evaluate the possible use of LFSR sequences generated by good seeds as codes to build shaft encoders with error-detecting and error-correcting capabilities, the following data was computed. For each value of e and $k \in \{2, 3\}$, the k-th Hamming complexity of the sequences generated by all the good seeds obtained for e in the ten executions of SearchGoodSeed $\left(e, 2, 10^{-2}\right)$ was computed. The minimum of these complexities is denoted by h_e^k. For $k = 2$ a finite Hamming complexity was obtained for every value of e, and for $k = 3$ we obtained $h_e^3 = \infty$ for 0.2% of the values of e in the range considered. This means that for a few values of e none of the good seeds found was able to generate a sequence with Hamming complexity at least 3. When considering the sequences $\left(h_e^2\right)$ and $\left(h_e^3\right)$, this infinite values were dropped. From (2) lower bounds for h_e^2 and h_e^3 can be computed. For $k = 2$ these bounds coincide with t_e, and for $k = 3$ they are denoted by b_e^3. Ploted in Fig. 4 one can find the sequences of means $\left(\bar{h}_e^2\right)$, $\left(\bar{h}_e^3\right)$ and $\left(\bar{b}_e^3\right)$. Note that sequences of means obtained from the data have the same log-like shapes as their lower bounds. Furthermore, in 91% of the cases we found $h_e^2 \leq 2t_e$, and in more than 98% of the cases $h_e^3 \leq 2b_e^3$. In this last bound, cases where $h_e^3 = \infty$ were considered too.

```
100010001111010001100011011000101011000011001
101010101101011001000001010000001001001011101
111011101001001000000101000001001101011010101
011001100001101010001101100011000101111000100
011101110000101110011100100111010100111100110
010101010010100110111110101111110110110100010
000100010110110111111010111110110010100101010
100110011110010101110010011100111010000111011
```

Fig. 5. A binary sequence S' with $|S'| = 360$ and $H_3(S') = 17$.

These results show that for every e in the range considered it is possible to build an absolute encoder with the capability to detect 1 error with a reasonable number of detectors. Furhtermore, for almost every e in the range one can construct sequences to label shaft encoders wich can detect 2 errors or correct 1 error using a few more detectors. In fact, the mean of the quotients h_e^3/h_e^2 equals 1.2.

As a particular example we consider the case $e = 360$. This corresponds to the case where one wants to build a shaft encoder with a resolution of 1^{o}. In this case the theoretical lower bound for the number of sensors is $\lceil \log_2 360 \rceil = 9$ and the number of sensors required by the standard seed is 15. Using SearchGoodSeed $\left(9, 2, 10^{-2}\right)$ ten times, a seed generating a sequence with combinatorial complexity equal to 10 was obtained. From these calls ten good seeds were found. Among those, one seed was able to generate a sequence S with $H_2(S) = 15$ and another seed generated a sequence S' with $H_3(S') = 17$. This last sequence is shown in Fig. 5.

V. Conclusion

In this paper the problem of building single-tracked absolute shaft encoders was studied. An approach using non-maximal LFSR sequences was considered. In particular, the number of detectors required by the methods presented in [11], [12] was studied and shown to be, on average, exponentially away from the theoretical lower bound. A variation on this method was proposed in order to overcome this limitation. We identified a set of good seeds that can, in principle, be used to generate sequences with small combinatorial complexity. A stochastic search algorithm to explore the space of these seeds was proposed and experimentally evaluated. The results of this evaluation show that the proposed method is able to bridge the exponential gap and obtain sequences that use a number of detectors that is, on average, within a constant factor of the lower bound.

Furthermore, the possibility to embed error-detecting and error-correcting capabilites inside the codes generated by LFSRs with these good seeds was explored. The number of detectors needed to pursue this goal has been evaluated experimentally and shown to be close (in the same average sense as before) to its theoretical lower bound. Therefore, it is possible to build single-tracked absolute encoders of arbitrary resolution which are robust against reading errors, and this can be done by only moderately increasing the number of detectors. To the best of our knowledge, this is the first time that such an approach is considered in the literature.

Many issues still need to be addressed before this method can be implemented in practice. In particular, efficient error-

correcting algorithms for these codes need to be developed. Note that a naïve table look-up algorithm would work with time $O(n)$ but would require space $O(q^n)$. Other possible extensions to this work can be considered. In particular, developing a theory of error correcting codes based on LFSR sequences capable of answering questions like 'for which values of e can certain bounds be attained?' or 'is there another class of seeds that generates sequences with smaller Hamming complexities?'. Furthermore, it would be worth investigating if assumption 3 or another similar statement can be proved or disproved. This would provide a solider theoretical ground in which stochastic searches like the one proposed here can be formally justified. From the implementation point of view, an open question is whether detectors arranged in a cleverer way along LFSR sequences can improve any aspect about the encoder. And if so, how and to what extent.

APPENDIX
PROOF OF LEMMA 1

The proof relies heavily on notation and results from [12]. In particular, recall that given two coprime integers, e and q, the order of q modulo e, denoted by $\mathrm{ord}_q(q)$, is the smallest positive integer i such that $q^i \equiv 1 \mod e$. One has $\mathrm{ord}_e(q) \leq e-1$. Note also that if $a \geq 2$ and $b \geq 3$, then $a+b \leq ab-1$.

Proof of lemma 1: Let us write $e = p^{\alpha_0} p_1^{\alpha_1} \cdots p_t^{\alpha_t} = p^{\alpha_0} e_*$ with $t \geq 0$, $\alpha_0 \geq 0$ and the other $\alpha_i \geq 1$. Note that e_* and p are coprime. Three different cases will be considered. Assume first that $t = 0$ and let $s = p^{\alpha_0 - 1} + 1$. Then $a(x) = (x-1)^s$ corresponding to form (4) with $k = 0$. Since $p \geq 2$, we get $\deg(a) = e/p + 1 \leq e-1$. Now assume that $t > 0$ and $\alpha_0 = 0$. This correponds to form (3), $a(x) = a_1(x) \cdots a_k(x)$, with $k \geq 1$. The factors satisfy $\deg(a_i) = \mathrm{ord}_{e_i}(q)$ where e_i are pairwise coprime integers such that $e = \prod e_i$. Then

$$\deg(a) = \sum \mathrm{ord}_{e_i}(q) \leq \sum (e_i - 1) \leq e - 1.$$

Finally, let us consider the general case correponding to form (4) with $k > 0$ and $s > 1$. In this case one has that $\deg(a_i) = \mathrm{ord}_{e_i}(q)$ and $\prod e_i = e_*$, where $1 \leq i \leq k+1$. Since $s = p^{\alpha_0 - 1} + 1$, by (4) in [12] and the previous case we have

$$\deg(a) \leq \sum \mathrm{ord}_{e_i}(q) + s \leq e_* + p^{\alpha_0} \leq e - 1.$$

■

REFERENCES

[1] F. Baronti, F. Lenzi, R. Roncella, R. Saletti, and O. D. Tanna, "Electronic control of a motorcycle suspension for preload self-adjustment," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 7, 2008.

[2] R. Jared, A. Arthur, S. Andreae, A. Biocca, R. Cohen, J. Franck, J. Fuertes, O. Gabor, J. Llacer, T. Mast, J. Meng, T. Merrick, R. Minor, J. Nelson, M. Orayani, P. Salz, B. Shaefer, and C. Witebsky, "The W.M. KECK telescope segmented primary mirror active control system," in *SPIE Symp. on Astronomical Telescopes & Instrumentation for the 21st. Century Proceedings*, 1990.

[3] N. Medrano-Marques, G. Zatorre-Navarro, and S. Celma-Pueyo, "A tunable analog conditioning circuit applied to magnetoresistive sensors," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 2, 2008.

[4] I. J. Good, "Normal recurring decimals," *J. London Math.*, vol. 21, 1946.

[5] B. Arazi, "Position recovery using binary sequences," *Electronics Letters*, vol. 20, 1984.

[6] G. H. Tomlinson, "Abolute-type shaft encoder using shift register sequences," *Electronics Letters*, vol. 23, 1987.

[7] E. M. Petriu, "Absolute position measurement using pseudo-random binary encoding," *IEEE Instrumentation and Measurement Magazine*, vol. 1, no. 3, 1998.

[8] A. P. Hiltgen, K. G. Paterson, and M. Brandestini, "Single-track gray codes," *IEEE Transactions on Information Theory*, vol. 42, no. 5, 1996.

[9] M. Schwarts and T. Etzion, "The structure of single-track gray codes," *IEEE Transactions on Information Theory*, vol. 45, no. 7, 1999.

[10] E. M. Petriu, "Abolute-type pseudorandom shaft encoder with any desired resolution," *Electronics Letters*, vol. 21, 1985.

[11] J. M. Fuertes, B. Balle, and E. Ventura, "Absolute-type shaft encoding using LFSR sequences with a prescribed length," *IEEE Trans. Instrumentation and Measurement Magazine*, vol. 57, no. 5, 2008.

[12] B. Balle, E. Ventura, and J. M. Fuertes, "An algorithm to design prescribed length codes for single-tracked shaft encoders," in *Proceedings of the 2009 IEEE International Conference on Mechatronics*, 2009.

[13] J. H. van Lint, *Introduction to coding theory*. Springer, 1999.

[14] A. Lempel, "m-ary closed sequences," *Journal of Combinatorial Theory*, vol. 10, 1971.

[15] E. Ventura, "Dynamic structure of matrices over finite fields," in *Proceedings of EAMA-97*, 1997.