Construction of Efficient q-ary Balanced Codes

Elie N. Mambou and Theo G. Swart
Dept. of Electrical and Electronic Engineering Science, University of Johannesburg
P. O. Box 524, Auckland Park, 2006, South Africa
Email: {emambou, tgswart}@uj.ac.za

***Abstract*—Knuth proposed a simple scheme for balancing codewords, which was later extended for generating q-ary balanced codewords. The redundancy of existing schemes for balancing q-ary sequences is larger than that of the full balanced set which is the minimum achievable redundancy. In this article, we present a simple and efficient method to encode the prefix that results in less redundancy for the construction of q-ary balanced codewords.**

Index Terms*—*balanced codes, balancing point, redundancy, q-ary alphabet, parallel decoding scheme, Knuth's scheme.

I. Introduction

It is undeniable that balanced or dc-free q-ary codes are widely used in magnetic and optical recording, detection of unidirectional errors, noise reduction in VLSI design systems, transmission of high power through cables. Also, it was established that q-ary balanced sequences can be used in various communication systems, for example, in visible light communication (VLC), they can be used to balance the flickering and perform dimming of lighting as well as compensate average color in color shift keying (CSK).

A simple and efficient scheme was proposed by Knuth [1] to generate binary balanced codewords; this consists of inverting bits within a sequence up to a certain index referred to as the balancing point. This balancing point is encoded as a prefix for enabling recovering of the original information at the receiver end.

A scheme was proposed in [2] to compress the redundancy of binary balanced codewords based on the multiplicity of balanced encoded codewords. This method was not successful enough and then a further improvement was made by Immink and Weber [3] through a very efficient scheme that decreased the redundancy of Knuth's traditional algorithm from $\log_2 n$ to $\log_2 \frac{n}{2} + 1$, with n being the length of the information sequence. A modification of this method was presented in [4] for packet transmission systems with a prefix length of $\log_2 \frac{n}{2}$. This is the same result obtained by Al-Rababa's *et al.* in [5] by exploiting the multiplicity of balancing points within a sequence as introduced in [2].

Many results have been published to improve Knuth's binary parallel balancing scheme. In [6], a method for parallel decoding of q-ary codes was presented using r check digits and $\frac{q^r-1}{q-1}$ source digits. Swart and Weber [7] proposed another balancing scheme for q-ary sequences of length n with parallel decoding, with a prefix length of $\log_q n$ for very long sequences. This scheme was extended by Mambou and Swart in [8] to provide a complete encoding and decoding method with Gray sequences as prefixes achieving a redundancy of $\log_q n + 2$, resulting in an overall (encoded information and prefix) balanced codeword.

In this paper, we propose an efficient q-ary balancing method based on [3] and [7] which provides reduce the redundancy of the resulting code.

The rest of this paper is organized as follows: some necessary background is discussed in Section II. The efficient encoding of q-ary sequence is presented in Section III, followed by the decoding in Section IV. Section V outlines some performance analysis as well as discussions of the proposed scheme. Finally the paper is concluded in Section VI.

II. Background

Let $\boldsymbol{x} = (x_0 x_1 \dots x_{k-1})$ be a q-ary sequence of length k; $\boldsymbol{p} = (p_0 p_1 \dots p_{r-1})$, the prefix of length r to be appended to $\boldsymbol{x}$. The codeword, $\boldsymbol{c} = (c_0 c_1 \dots c_{n-1})$ of length $n = k + r$ is the transmitted codeword made of the encoding of $\boldsymbol{x}$ appended with $\boldsymbol{p}$, $\boldsymbol{c} = (\boldsymbol{x}|\boldsymbol{p})$. All these sequences are defined within the alphabet set $\mathcal{Q} = \{0, 1, \dots, q-1\}$. $w(\boldsymbol{x})$ refers to the algebraic sum of symbols within $\boldsymbol{x}$. The q-ary codeword $\boldsymbol{x}$ is said to be balanced if

$$w(\boldsymbol{x}) = \sum_{i=0}^{k-1} x_i = \frac{k(q-1)}{2}.$$

We denote $\beta_{(k,q)}$ as the balancing value of a q-ary sequence of length k. For the rest of this paper, the assumption is made that $\beta_{(k,q)}$ is always an integer value, that is k must not be odd while q is even.

A polar representation of $\boldsymbol{x} = (x_0 x_1 \dots x_{k-1})$ is as follows: $x_i \in \{-(q-1), \dots, -2, -1, +1, +2, \dots, +(q-1)\}$ for q even and $x_i \in \{-(q-1)/2, \dots, -1, 0, +1, \dots, +(q-1)/2\}$ for odd q.

This previous notation allows us to define the *running digital sum* (RDS) of the first t symbols with $t \leq k$ for q-ary sequences as

$$z_t(\boldsymbol{x}) = \sum_{i=1}^{t} x_i.$$

The RDS through the whole length k is also referred to as the *imbalance*; the word $\boldsymbol{x}$ is balanced if and only if $z_k(\boldsymbol{x}) = 0$.

Let Q_q^k denote the full set of q-ary sequences and $C(Q_q^k) = |Q_q^k|$, the cardinality of that set. The values of $C(Q_q^k)$ correspond to the central q-nomial coefficients of $(1 + \alpha + \alpha^2 + \alpha^3 + \dots + \alpha^{q1})^k$, where α is a variable. All these values can be obtained online from [9].

TABLE I
CARDINALITIES FOR FULL BALANCED SETS

	$q=2$	$q=3$	$q=4$	$q=5$	$q=6$
$k=3$		7		19	
$k=4$	6	19	44	85	146
$k=5$		51		381	
$k=6$	20	141	580	1751	4332
$k=7$		393		8135	
$k=8$	70	1107	8092	38165	135954
$k=9$		3139		180325	
$k=10$	252	8953	116304	856945	4395456

From [10] and [7], it was derived that,

$$C(Q_q^k) = q^k \sqrt{\frac{6}{\pi k(q^2-1)}} \Big(1 + \mathcal{O}\Big(\frac{1}{k}\Big)\Big). \quad (1)$$

By applying the logarithm, it follows that

$$\log_q(C(Q_q^k)) \approx k - \frac{1}{2}\log_q k - \frac{1}{2}\log_q \frac{\pi}{6} - \frac{1}{2}\log_q(q^2-1). \quad (2)$$

Table I presents the cardinalities of full balanced sets for $k \in [3-10]$ and $q \in [2-6]$.

A. Balancing of q-ary sequences

It was established in [7] that any q-ary sequence of length k can be balanced through a scheme that consists of adding modulo q a set of kq balancing sequences $\boldsymbol{b}_{s,p}$ to the original sequence $\boldsymbol{x}$.

The balancing sequence $\boldsymbol{b}_{s,p} = (b_0 b_1 \ldots b_{k-1})$ of length k is evaluated as follows:

$$b_i = \begin{cases} s, & i > p, \\ s+1 \pmod q, & i \le p, \end{cases}$$

where s, p are integers such that $0 \le s \le q-1$ and $0 \le p \le k-1$. z denotes the iterator through all possible kq balancing sequences, $z = sk + p$ with $0 \le z \le kq-1$.

If $\boldsymbol{y}$ are the resulting sequences obtained by adding $\boldsymbol{x}$ distinctly to the kq $\boldsymbol{b}_{s,p}$ as $\boldsymbol{y} = \boldsymbol{x} \oplus_q \boldsymbol{b}_{s,p}$, then this process will always lead to at least one occurrence of a balanced sequence $\boldsymbol{y}$, according to [7].

Example 1 Consider the ternary sequence, 2102 of length $k = 4$ to be balanced.

z	$\boldsymbol{x}$	$\oplus_q$	$\boldsymbol{b}_z$	$=$	$\boldsymbol{y}$	$w(\boldsymbol{y})$
0	(2102)	$\oplus_3$	(0000)	$=$	(2102)	5
1	(2102)	$\oplus_3$	(1000)	$=$	(0102)	3
2	(2102)	$\oplus_3$	(1100)	$=$	(0202)	**4**
3	(2102)	$\oplus_3$	(1110)	$=$	(0212)	5
4	(2102)	$\oplus_3$	(1111)	$=$	(0210)	3
5	(2102)	$\oplus_3$	(2111)	$=$	(1210)	**4**
6	(2102)	$\oplus_3$	(2211)	$=$	(1010)	2
7	(2102)	$\oplus_3$	(2221)	$=$	(1020)	3
8	(2102)	$\oplus_3$	(2222)	$=$	(1021)	**4**
9	(2102)	$\oplus_3$	(0222)	$=$	(2021)	5
10	(2102)	$\oplus_3$	(0022)	$=$	(2121)	6
11	(2102)	$\oplus_3$	(0002)	$=$	(2101)	**4**

For this case, there are 4 distinct ways of balancing the information sequence 2102. □

In analogy with Knuth's scheme that complements digits up to $(k-1)$-th index, a q-ary sequence of length k always has kq different ways of being complemented corresponding to the kq balancing sequences. Let e be an index leading to a balanced q-ary sequence and $\boldsymbol{x}^e$ be the codeword obtained after complementing the first e symbols of $\boldsymbol{x}$, with $0 \le e \le kq-1$.

In Example 1, $\boldsymbol{x}^2 = (0202)$, $\boldsymbol{x}^5 = (1210)$, $\boldsymbol{x}^8 = (1021)$ and $\boldsymbol{x}^{11} = (2101)$ are the balanced 3-ary sequences obtained from the information sequence 2102. The values of e equals the corresponding iterator z. This construction may lead to many occurrences of balanced codewords as it is the case in this example. It was showed in [2] that the multiplicity of these balanced occurrences may be used to transmit auxiliary data and then reduce the redundancy in the binary case.

B. Efficient binary balanced sequences

The efficient encoding scheme presented in [3] consists of associating or mapping every sequence $\boldsymbol{x}$ from the set of all binary sequences of length k, of cardinality 2^k to a balanced one denoted as $\boldsymbol{x}'$ within the set of balanced codewords, of cardinality $\binom{k}{k/2}$ as will be illustrated in Example 2.

The prefix of the encoded sequence corresponds to the rank of the information sequence, $\boldsymbol{x}$ within the subset of all sequences leading to the same balanced codeword, $\boldsymbol{x}'$. It was showed that the size of that subset, $s(\boldsymbol{x}')$ is such that: $2 \le s(\boldsymbol{x}') \le \frac{k}{2} + 1$, where $s(\boldsymbol{x}') = \max\{z_t(\boldsymbol{x})\} - \min\{z_t(\boldsymbol{x})\} + 1$ with $\max\{z_t(\boldsymbol{x})\}$ and $\min\{z_t(\boldsymbol{x})\}$ being the maximum and minimum RDS value of $\boldsymbol{x}$ respectively.

Example 2 Consider all binary sequences of length 4; the cardinality of this set is $2^4 = 16$ and there are $\binom{4}{2} = 6$ balanced sequences out of the 16.

$\boldsymbol{x}'$	0011	0101	0110	1001	1010	1100	$\boldsymbol{p}$
$\boldsymbol{x}$	①1011	1101	1000	0001	0010	0000	00
	②1111	1010	1110	0111	0101	0100	01
	③1100		1001	0110		0011	10

(3)

Lines ①, ② and ③ show the balancing process according to Knuth's algorithm:
①1011 $\rightarrow$ 0011
②1111 $\rightarrow$ 0111 $\rightarrow$ 0011
③1100 $\rightarrow$ 0100 $\rightarrow$ 0000 $\rightarrow$ 0010 $\rightarrow$ 0011.

(3) presents the mapping of all information sequences into balanced sequence subsets as described in [3] for binary sequences of length 4. $\boldsymbol{p}$ represent prefixes.

For instance the encoding of the information sequence $\boldsymbol{x} = (1110)$ gives 0110**01**, where the bold part represents the appended prefix and the cardinality of $s(0110)$ is $|s(0110)| = 3$. □

In [4], a modified version of [3] was proposed for packet-based transmission. It was observed that any balanced codeword is always associated with another balanced one. Therefore balanced sequences were excluded from sets of information sequences as in [3]; this important observation leads to the compression of $s(\boldsymbol{x}')$ by removing the already balanced sequences.

Example 3 Considering all binary sequences of length 4 as in Example 2

$\boldsymbol{x}'$	0011	0101	0110	1001	1010	1100	$\boldsymbol{p}$
$\boldsymbol{x}$	1011	1101	1000	0001	0010	0000	0
	1111		1110	0111		0100	1

(4)

We see that prefixes can be represented only with 1 bit instead of 2 like previously. □

III. Encoding of q-ary sequences

The scheme presented in [7] stated that the lower bound for the number of position indexes where a q-ary sequence can be balanced is 1. The following theorem 1 provides the upper bound for these balanced indexes.

Theorem 1 *There are at most k indexes where a q-ary sequence of length k can be balanced.* □

Proof: Let e be the index where a balanced sequence is obtained with $0 \leq e \leq kq - 1$. One can observe that at the neighboring indexes $e + (q-1)$ or $e - (q-1)$, a balanced sequence can not arise. We conclude that the possible number of indexes where balancing can occur is at most k. ■

The proposed construction consists of finding the associated balanced sequence to an information sequence by applying the scheme in [7] and then determining the cardinality of this subset and finally encoding the prefix as the rank occupied by $\boldsymbol{x}$ within that subset.

By applying the scheme in [7] to find the least index to balance a q-ary sequence and then associate it with a balanced codeword within the set of balanced q-ary codewords of length k as it is done in [3] for binary sets. This is shown in the next example.

In examples 4 to 8, we tabulate the information sequences into subsets such that each unbalanced sequence is associated with a balanced codeword.

Example 4 Consider all ternary sequences of length 3; the cardinality of this set is $3^3 = 27$ and there are 7 balanced sequences out of the 27.

$\boldsymbol{x}'$	012	021	102	111	120	201	210	$\boldsymbol{p}$
$\boldsymbol{x}$	201	210	002	000	010	101	100	00
	202	211	021	001	012	120	102	01
	212	221	022	011	020	121	110	02
				111				10
				112				10
				122				11
				200				20
				220				21
				222				22

□

Example 5 For all ternary sequences of length 4; there are 19 balanced ones out of the $3^4 = 81$.

$\boldsymbol{x}'$	0022	0112	0121	0202	0211	0220	$\boldsymbol{p}$
$\boldsymbol{x}$	1111	2001	2010	2102	1022	1012	00
	2212	2002	2011	1121	2100	2022	01
	2222	2012	2021	2122	2101	2110	02
		2112	2121	2202	2111	1112	10
					2211	2120	11
						2220	12

1012	1021	1102	1111	1120	1201	1210	$\boldsymbol{p}$
0012	0021	0002	0000	0010	0101	0100	00
0202	0211	0022	0001	0020	0121	0110	01
0212	0221	0102	0011	0120	0201	0210	02
2201	1102	2221	0111		2020		10
	2210		2000				11
			2200				12

2002	2011	2020	2101	2110	2200	$\boldsymbol{p}$
1202	1011	1020	1001	1000	1002	00
1221	0200	0112	1021	1010	1100	01
1222	1201	1210	1101	1110	0122	02
		1220			1122	10
					1200	11

□

Example 6 For all 5-ary sequences of length 3; there are 19 balanced ones out of the $5^3 = 125$.

$\boldsymbol{x}'$	024	033	042	114	123	132	141	204	$\boldsymbol{p}$
$\boldsymbol{x}$	303	312	321	003	013	021	030	033	00
	313	322	331	004	023	022	031	043	01
	413	422	431	014	123	032	041	104	02
	414	423	432	332	234	304	140	130	03
	424	433	442	403	340	410	213	143	04
				443	401	411	324	144	10
					402	421	420		11
					412		430		12

213	222	231	240	303	312	321	330	$\boldsymbol{p}$
022	000	010	024	132	040	044	114	00
042	001	020	034	142	101	100	220	01
102	011	120	134	203	141	110	230	02
103	111	121		243	201	210		03
113	112	131			202	211		04
330	122	343			212	221		10
441	222	404			323			11
	223				434			12
	233							13
	333							14
	334							20
	344							21
	400							22
	440							23
	444							24

402	411	420	$\boldsymbol{p}$
124	133	204	00
241	200	214	01
231	240	310	02
241	244	314	03
302	300	320	04
	311		10

□

Example 7 For all 5-ary sequences of length 2; there are 5 balanced ones out of the $5^2 = 25$.

$\boldsymbol{x}'$	04	13	22	31	40	$\boldsymbol{p}$
$\boldsymbol{x}$	22	02	00	04	13	0
	32	03	01	10	23	1
	33	31	11	14	24	2
	43	41	12	20	30	3
	44	42	40	21	34	4

□

Example 8 For all 6-ary sequences of length 2; there are 6 balanced ones out of the $6^2 = 36$.

$\boldsymbol{x}'$	05	14	23	32	41	50	$\boldsymbol{p}$
$\boldsymbol{x}$	32	03	01	00	14	23	0
	33	04	02	05	15	24	1
	43	41	12	10	20	34	2
	44	42	13	11	25	35	3
	54	52	50	21	30	40	4
	55	53	51	22	31	45	5

□

We can categorize encoding into three cases depending on parameters k and q. The case when the value k and q are odd as in Examples 4 and 7; the case that k is even while q is odd as in Examples 5 and 6; and the case where k and q are even as in Examples 2 and 8. This categorization is based on the distribution of the size $s(\boldsymbol{x}')$ across balanced sequences $\boldsymbol{x}'$.

Theorem 2 *The cardinality of sets of associated q-ary sequences with balanced codewords is such that:*

- *For k even and q even, $|s(\boldsymbol{x}')| \leq q(k-1)$;*
- *For k even and q odd, $|s(\boldsymbol{x}')| \leq kq/2$; and*
- *For k odd and q odd, $|s(\boldsymbol{x}')| \leq kq$.* □

In order to prove Theorem 2, we should know the balanced codewords that have the maximum $|s(\boldsymbol{x}')|$ given a codebook, so that we can compute its bounds. We observed a specific structure on balanced codewords having maximum $|s(\boldsymbol{x}')|$. Depending of the parity of k and q, they have a structure made of symbols $\frac{q-1}{2}$ and/or $q-1$.

Lemma 1 *Balanced codewords having maximum $|s(\boldsymbol{x}')|$ are as follows:*

- *For q odd, the structure is*

$$\underbrace{\left(\frac{q-1}{2}\right)\cdots\left(\frac{q-1}{2}\right)}_{k\ times} \tag{5}$$

- *For q even, the structure is*

$$\underbrace{(q-1)\ldots(q-1)}_{k/q\ times}\underbrace{\left(\frac{q-1}{2}\right)\cdots\left(\frac{q-1}{2}\right)}_{k\text{-}k/q\ times} \tag{6}$$

□

SKETCH OF PROOF (Lemma 1): It is known that the q-ary balancing technique according to [7] follows a $(1, q-1)$-random walk given every information sequence.

- For q odd, it is observed that balanced codewords following structure (5) are located almost at the centre of the full set of balanced codewords in the lexicographic order; this means that, on average, most of the information sequences are associated with those balanced codewords while following the $(1, q-1)$-random walk. It can be verified that the sum of symbols in structure (6) equals the balancing value of $k(q-1)/2$.
- For q even, information sequences are balanced through the process of $\boldsymbol{x} \oplus_q \boldsymbol{b}(s,p)$; on average, they are associated with balancing codewords following the structure of (6).
Similarly, the sum of symbols in structure (6) also equals the balancing value of $k(q-1)/2$. ■

SKETCH OF PROOF (Theorem 2): This proof becomes trivial as we know exactly the balanced codeword with the maximum $|s(\boldsymbol{x}')|$ as per Lemma 1. We have to count the number of sequences associated with the balanced codeword with the maximum $|s(\boldsymbol{x}')|$ and find the upper bound given the parity of k and q. On average, each subset $s(\boldsymbol{x}')$ is such that $|s(\boldsymbol{x}')| = \frac{q^k}{|\mathcal{S}_q^k|}$, where $|\mathcal{S}_q^k|$ is the cardinality of the full set of balanced q-ary sequences of length k.

Each q-ary sequence of length k, $\boldsymbol{x}$, in $s(\boldsymbol{x}')$, is associated with the balanced codeword $\boldsymbol{x}'$, and has a unique $\boldsymbol{b}(s,p)$ within each $s(\boldsymbol{x}')$. This implies that the $|s(\boldsymbol{x}')|$ is upper bounded by the cardinality of balancing sequences which equals kq.

- For k even and q even, the balanced codeword having the maximum $|s(\boldsymbol{x}')|$ has the structure (6). Due to the fact that balancing an information sequence $\boldsymbol{x}$ can lead to several balanced states and that only the first one is considered, only the first $q(k-1)$ balancing sequences are necessary to balance any $\boldsymbol{x}$.
- For k even and q odd, the balanced codeword having the maximum $|s(\boldsymbol{x}')|$ has the structure (5). Because of the parity constraint, only the first $kq/2$ balancing sequences are necessary to balance any $\boldsymbol{x}$.
- For k odd and q odd, the balanced codeword having the maximum $|s(\boldsymbol{x}')|$ has the structure (5). All kq balancing sequences are required to ensure that every $\boldsymbol{x}$ is balanced. ■

After differentiating these three cases of scenario, we define the steps for our encoding scheme:

- Given a random q-ary sequence $\boldsymbol{x}$ to be balanced, find the corresponding balanced sequence $\boldsymbol{x}'$.
- Then find all information sequences associated with $\boldsymbol{x}'$.
- Rank all the elements of this subset into lexicographic order.
- Finally, the rank of the information sequence $\boldsymbol{x}$ is encoded as the prefix.

IV. DECODING

The decoding process is as follows:

- The prefix is extracted from the overall received codeword of length n.
- Then all information sequence candidates associated with $\boldsymbol{x}'$ are listed and ordered lexicographically.
- Finally, the prefix is mapped to the rank of the right original information sequence.

TABLE II
PROPOSED SCHEME REDUNDANCIES

k	q	k vs. r
Even	Even	$k = q^r + 1$
Even	Odd	$k = 2q^r$
Odd	Odd	$k = q^r$

This construction is similar for the three cases of scenario depending on parameters k and q as defined previously. For the case where k and q are even, the size of the subset is the most efficient, $|s(\boldsymbol{x'})| = kq/2$; this decreases the redundancy of the prefix and improves efficiency.

Example 9 We would like to retrieve the original sequence from the received codeword, $(\mathbf{11}11000011)$, where the bold symbols is the prefix.

Sequences	Prefix rank
01000011	0 (**0**)
00000011	1 (**1**)
00110011	2 (**10**)
00111011	3 (**11**)
00111111	4 (**100**)

(7)

(7) presents all information sequence candidates associated with the balanced codeword 11000011 with corresponding prefix ranks.

Therefore, the received codeword **11**11000011 is mapped to the original information sequence, 00111011. □

V. ANALYSIS AND DISCUSSIONS

The construction in [7] has the following information length k:

$$k \leq \frac{\mathcal{S}_q^r}{q} \approx q^{r-1}\sqrt{\frac{6}{\pi r(q^2-1)}}. \quad (8)$$

In [11], two schemes are presented for k information symbols, where the first one satisfies the bound

$$k \leq \frac{q^r - 1}{q-1}, \quad (9)$$

and the second one is

$$k \leq 2\frac{q^r - 1}{q-1} - r. \quad (10)$$

The prefix-less scheme presented in [12] shows that

$$k \leq q^{r-1} - r. \quad (11)$$

Two constructions with parallel decoding are presented in [6]. The first construction, where the prefixes are also balanced as in [7], has its information length as a function of r as

$$k \leq \frac{\mathcal{S}_q^r - \{q \bmod 2 + [(q-1)k] \bmod 2\}}{q-1}. \quad (12)$$

The second construction, where the prefixes may not be balanced, is a refinement of the first and has an information length the same as (10).

Information length versus redundancies functions of the proposed construction are presented in Table II. The equations from this table were obtained and verified after many simulations.

Fig. 1 presents the comparison of the information lengths, k versus redundancies, r, for some existing constructions as discussed above for $q = 4$, 32 and 128. It can be observed that for large values of q, the proposed scheme when k is even and q odd is only comparable to that from [11] as per (9); the two other proposed schemes have a performance comparable only to [12] and [11] as per (11) and (10) respectively. Moreover, the proposed scheme becomes much less redundant for large q.

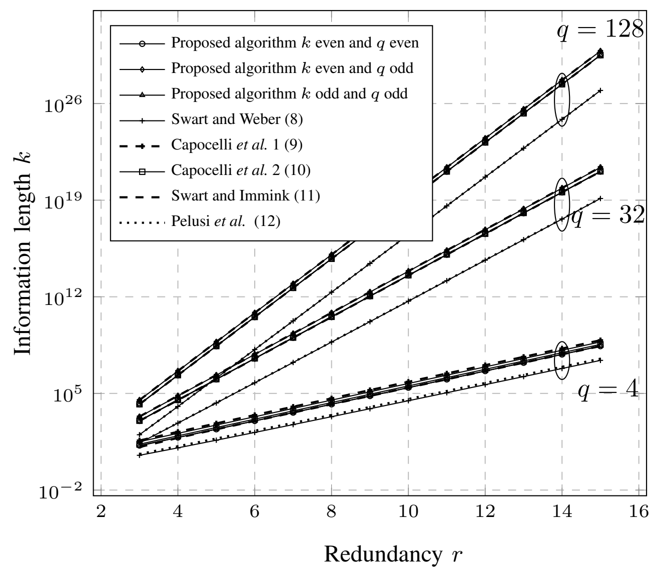

Fig. 1. Redundancy comparison

In [8], a construction was described by Mambou and Swart for full balancing q-ary sequence; a full balancing scheme refers to a construction that achieve balancing of information and prefix together; it was stated that

$$k = q^{r'-2}. \quad (13)$$

Where r' represents the redundancy required to balance the overall frame made of encoded source and prefix together. Table III presents the redundancies of the proposed scheme based on the full balancing construction presented in [8].

TABLE III
PROPOSED SCHEME REDUNDANCIES

k	q	Redundancy r'	k vs. r'
Even	Even	$r' = r + \log_q rq$	$k = q^{r'} + 1$
Even	Odd	$r' = r + \log_q rq$	$k = 2q^{r'}$
Odd	Odd	$r' = r + \log_q rq$	$k = q^{r'}$

Fig. 2 shows the proposed construction performance against that from [8] as per (13);

It can be observed that, the proposed scheme based on the full construction of [8] presents a considerable improvement in redundancies compared to [8]. This implies that the proposed algorithm can make an efficient construction for full balancing of q-ary sequences in fixed length as it outperforms several state-of-the-art schemes.

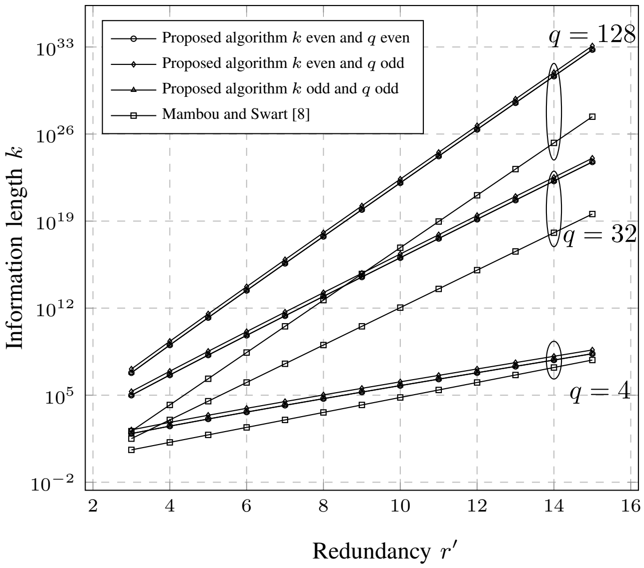

Fig. 2. Redundancy comparison

VI. Conclusion

A simple and efficient construction was presented to generate balanced q-ary codes. The proposed method is a fixed length scheme based on the parity of parameters k and q; it is fast, efficient and less redundant than other schemes for some cases and the decoding is done in parallel. Additionally, it does not make use of look-up tables and it is suitable for both long and short length codes. Furthermore, the proposed scheme was integrated in the full balancing construction presented in [8] and the redundancy was also improved for some cases.

However, the distribution of the prefix length as well as the average efficiency should be computed and analyzed for the proposed construction.

VII. Acknowledgement

This work is based on research supported in part by the Global Excellence Stature program of the University of Johannesburg and the National Research Foundation of South Africa (Grant number 109543).

References

[1] D. E. Knuth, "Efficient balanced codes," *IEEE Transactions on Information Theory*, vol. 32, no. 1, pp. 51–53, Jan. 1986.

[2] J. H. Weber and K. A. S. Immink, "Knuth's balancing of codewords revisited," *IEEE International Symposium on Information Theory,* Toronto, Canada, Jul. 6–11, 2008, pp. 1567–1571.

[3] K. A. S. Immink and J. H. Weber, "Very efficient balanced codes," *Journal on Selected Areas in Communications*, vol. 28, no. 2, pp. 188-192, Feb. 2010.

[4] E. N. Mambou, E. Esenogho and H. C. Ferreira, "Improving the redundancy of Knuth's balancing scheme for packet transmission systems," to appear in *Turkish Journal of Electrical Engineering and Computer Sciences*, 2019.

[5] A. Al-rababa'a, D. Dube and J. Y. Chouinard, "Using bit recycling to reduce Knuth's balanced codes redundancy," *Canadian Workshop on Information Theory*, Toronto, ON, Canada, Jun. 18-21, 2013, pp. 6-11.

[6] D. Pelusi, S. Elmougy, L. G. Tallini and B. Bose, "m-ary balanced codes with parallel decoding," *IEEE Transactions on Information Theory*, vol. 61, no. 6, pp. 3251–3264, Jun. 2015.

[7] T. G. Swart and J. H. Weber, "Efficient balancing of q-ary sequences with parallel decoding," in *IEEE International Symposium in Information Theory*, Seoul, South Korea, Jun. 29–Jul. 3, 2009, pp. 1564–1568.

[8] E. N. Mambou and T. G. Swart, "A Construction for Balancing Non-Binary Sequences Based on Gray Code Prefixes," *IEEE Transactions on Information Theory*, vol. 64, no. 8, pp. 5961–5969, Aug. 2018.

[9] N. J. A. Sloane, "The on-line encyclopedia of integer sequences," 2005. [Online]. Available: https://oeis.org/

[10] Z. Star, "An asymptotic formula in the theory of compositions," *Aequationes Mathematicae,* vol. 13, pp. 279–284, 1975.

[11] R. M. Capocelli, L. Gargano and U. Vaccaro, "Efficient q-ary immutable codes," *Discrete Applied Mathematics*, vol. 33, pp. 25–41, 1991.

[12] T. G. Swart and K. A. S. Immink, "Prefixless q-ary balanced codes with ECC," in *2013 IEEE Information Theory Workshop (ITW)*, Sevilla, Spain, Sept. 9–13, 2013, pp. 1–5.