Block Codes for Asynchronous Data Transmission over a Noiseless q–ary Channel

Vladimir B. Balakirsky
Euler Institute for Discrete Mathematics and its Applications (EIDMA)
c/o Eindhoven University of Technology
P.O.Box 513, 5600 MB Eindhoven, The Netherlands
v.b.balakirsky@ele.tue.nl

Abstract

We describe a class of codes that can be effectively used when one of q^n vectors of length n has to be delivered to the receiver over a noiseless q–ary channel in asynchronous mode assuming that the latter one receives the transmitted vector with the delay $\tau \in \{0, \ldots, n-1\}$, unknown in advance, while all other received symbols are arbitrarily chosen. The codes are specified for any n by a regular algorithm, which is based on properties of ordered, oriented, rooted, binary trees, and have length $N \approx n + 2\log_q n$. We show that these codes can be used in such a way that encoding and decoding complexities are measured by linear functions of n.

1 Statement of the synchronization problem and formulation of the result

Suppose that $q \geq 2$ is given and denote

$$\mathcal{Q} \stackrel{\triangle}{=} \{0, \ldots, q-1\}.$$

We address the following problem.

– The sender has to deliver one of q^n vectors

$$\mathbf{X} = (X_0, \ldots, X_{n-1}) \in \mathcal{Q}^n$$

to the receiver over a noiseless q–ary channel. Therefore the sender forms a q–ary vector (codeword) of length $N > n$, denoted by $f(\mathbf{X}) \in \mathcal{Q}^N$, and sends it to the channel.

– The receiver receives the transmitted vector with the delay $\tau \in \{0, \ldots, n-1\}$, which is unknown in advance, i.e.,

$$(Y_\tau, \ldots, Y_{\tau+N-1}) = f(\mathbf{X}) \qquad (1)$$

and all other symbols are arbitrary, where $Y_0, Y_1, \ldots \in \mathcal{Q}$ denote symbols of the received sequence.

– The receiver must uniquely decode the pair $(\tau, \mathbf{X})$ for any received sequence.

Theorem. *For any $q \geq 2$ and n, there is an explicit construction of a q–ary code of length*

$$N = \begin{cases} n + \lceil \log_q n \rceil + \lceil \log_q (n+1) \rceil, & \textit{if } q \textit{ is even and } n \textit{ is odd} \\ n + 1 + \lceil \log_q (n+1) \rceil + \lceil \log_q (n+2) \rceil, & \textit{otherwise} \end{cases}$$

which brings a solution to the synchronization problem. Furthermore, complexities of encoding and decoding algorithms are measured by linear functions of n.

To prove the theorem we present the construction of a code that can be viewed as a result of adaptation of the coding algorithms for ordered, oriented, rooted, binary trees [1], [2]. In foregoing considerations these trees will be referred to as regular binary trees if each non–terminal node has exactly 2 sons and as non–regular binary trees if each non–terminal node has either 1 son or 2 sons.

The following lower bound :

$$N \geq n + \log_q n \qquad (2)$$

can be helpful to evaluate the statement of the theorem. Really, let us consider the set of vectors $\mathbf{Y} = (Y_0, \ldots, Y_{N+n-2}) \in \mathcal{Q}^{N+n-1}$. If $\mathbf{Y}$ is one of q^{n-1} vectors, for which (1) holds, then the decoder has to map this vector to the pair $(\tau, \mathbf{X})$ because of the requirement of unique decoding. There are nq^n pairs and we write $nq^n \cdot q^{n-1} \leq q^{N+n-1}$ since q^{N+n-1} is the total number of vectors, and (2) follows.

It is known [3], [4] that inequality (2) is asymptotically tight in a sense that there are codes for at least $\alpha_n(q)M/n$ messages, where $\alpha_n(q)$ tends to a constant as $n \to \infty$. We design a different class of block codes whose cardinalities are "slightly" smaller in a sense that, if $n \to \infty$, then $N/n \to 1$ as a function $\log n/n$, like for optimal codes. However these codes can be used with the encoding and decoding algorithms having complexities, which are linear in n, while implementation of the known encoding and decoding algorithms can be a difficult computational problem.

We begin with the presentation of some known results, then specify our code constructions and discuss the ideas included in their description.

2 Block codes without overlaps

Block codes without overlaps, which are oriented to the solution to the synchronization problem formulated above, were introduced by V. I. Levenshtein [3], [4]. The q–ary code $\mathcal{C}_N^*$ consisting of codewords of length N is called the "code without overlaps" if and only if

$$(c_0, \ldots, c_i) \neq (c'_{N-1-i}, \ldots, c'_{N-1})$$

for all $(c_0, \ldots, c_{N-1}), (c'_0, \ldots, c'_{N-1}) \in \mathcal{C}_N^*$ and $i = 0, \ldots, n-1$. One can easily understand that any code brings a solution to the synchronization problem if and only if it has this property.

A subclass of q–ary codes considered in [3] can be presented in the following way. Let all codewords of a code begin with r zeroes and do not "contain" r zeroes, i.e.,

$$(c_i, \ldots, c_{i+r-1}) \neq (0, \ldots, 0)$$

for all $\mathbf{c} \in \mathcal{C}_N^*$ and $i = 1, \ldots, N-r$. The decoding algorithm is very simple : the decoder searches for r sequential zeroes in the received sequence $\mathbf{y}$ and decides that the delay is $\tau \in \{0, \ldots, n-1\}$ if $(y_\tau, \ldots, y_{\tau+r-1}) = (0, \ldots, 0)$ and $(y_i, \ldots, y_{i+r-1}) \neq (0, \ldots, 0)$ for all $i = \tau+1, \ldots, \tau+n-r$. The analysis of this scheme is based on the results by H. B. Mann [5] who showed that the total number of binary sequences of length t, do not containing r consecutive zeroes, is equal to the integer, which is the closest one to

$$\frac{\rho^{r+1} - \rho}{\rho^{r+1} - (q-1)r} \cdot \rho^t$$

where ρ is the largest positive root of the equation

$$\rho^{r+1} - q\rho^r + q - 1 = 0.$$

If

$$r = \log_q \frac{(q-1)^2 n}{q \ln q}$$

then

$$|\mathcal{C}_N^*| \simeq \frac{\ln q}{q^{q/(q-1)}} \frac{q^N}{N}$$

The value of N is also close to n when $|\mathcal{C}_N^*| = q^n$ for finite values of n. In particular, if $q = 2$, then any string of length 500 can be delivered to the receiver with 512 bits and any string of length 1011 can be delivered to the receiver with 1024 bits in asynchronous mode. We will introduce a different class of codes, which allows us to deliver any of 494 and 1002 bits in the examples above since there are difficulties in the implementation of codes $\mathcal{C}_N^*$. The problem for $q = 2$ can be formulated as follows : we have to construct a one–to–one mapping between any binary vector of length n and a binary vector of length N having the property that it can be partitioned into substrings $1, 01, \ldots, 0^{r-1}1$ and vise versa. Solution to this problem is equivalent to a realization of the so–called variable–to–fixed and fixed–to–variable length source coding procedures.

3 Encoding and decoding algorithms for the case when q is even and n is odd

3.1 Encoding algorithm

- Given a vector $\mathbf{X} \in \mathcal{Q}^n$, denote

$$x_i \triangleq \begin{cases} 0, & \text{if } X_i \in \{0, \ldots, q/2-1\} \\ 1, & \text{if } X_i \in \{q/2, \ldots, q-1\} \end{cases}$$

for all $i = 0, \ldots, n-1$.

- Find $j \in \{0, \ldots, n\}$ such that the inversion of the first $j-1$ components of the vector $\mathbf{x} = (x_0, \ldots, x_{n-1})$ transforms it to the vector $\mathbf{x}' = (x'_0, \ldots, x'_{n-1})$ having $(n+1)/2$ zeroes and $(n-1)/2$ ones, i.e.,

$$x'_0 + \ldots + x'_{n-1} = (n-1)/2$$

where

$$x'_i \triangleq \begin{cases} x_i \oplus 1, & \text{if } i \in \{0, \ldots, j-1\} \\ x_i, & \text{if } i \in \{j, \ldots, n-1\}. \end{cases}$$

- Find $k \in \{0, \ldots, n-1\}$ such that the cyclic shift of the vector $\mathbf{x}'$ by k positions to the right transforms it to the vector $\mathbf{z} = (z_0, \ldots, z_{n-1})$ having *the dominating property*, i.e.,

$$z_0 + \ldots + z_{t-1} \geq t/2$$

for all $t = 1, \ldots, n-1$, where

$$z_i \triangleq \begin{cases} x'_{n+i-k}, & \text{if } i \in \{0, \ldots, k-1\} \\ x'_{i-k}, & \text{if } i \in \{k, \ldots, n-1\}. \end{cases}$$

- Denote

$$X'_i \triangleq \begin{cases} q-1-X_i, & \text{if } i \in \{0, \ldots, j-1\} \\ X_i, & \text{if } i \in \{j, \ldots, n-1\} \end{cases}$$

and

$$Z_i \triangleq \begin{cases} X'_{n+i-k}, & \text{if } i \in \{0, \ldots, k-1\} \\ X'_{i-k}, & \text{if } i \in \{k, \ldots, n-1\}. \end{cases}$$

Table 1: Example of the transformation $\mathbf{X} \to \mathbf{Z}$ for $q = 4$ and $n = 15$.

	$\mathbf{X} =$	011011300212323
	$\mathbf{x} =$	000000100101111
$j = 1$	$\mathbf{x}' =$	100000100101111
	$\mathbf{X}' =$	311011300212323
$k = 6$	$\mathbf{z} =$	101111100000100
	$\mathbf{Z} =$	212323311011300

- Send the vector $(Z_{n-1}, \ldots, Z_0)$ to the channel.
- Form the standard representation of the integer k with $\lceil \log_q n \rceil$ q–ary symbols and send it to the channel.
- Form the standard representation of the integer j with $\lceil \log_q(n+1) \rceil$ q–ary symbols and send it to the channel.

For the example, given in Table 1, the vector $\mathbf{Z}$ in the reverse order is 003110113323212, and the standard representations of k and j in the alphabet $\{0, 1, 2, 3\}$ are $k = 6 \to 12$, $j = 1 \to 01$. Thus, the vector

$$f(\mathbf{X}) = 0031101133232121201$$

of length $15 + 4 = 19$ is the transmitted codeword.

3.2 Decoding algorithm

Let us first introduce the following algorithm for searching for the vector, having the dominating property, in a binary sequence $\mathbf{y} = (y_0, y_1, \ldots)$. This algorithm will be referred to as *the* $(0,0,1) \to (*,*,0)$ *procedure* (see Table 2). The decoder scans the received sequence and maps

$$(0,0,1) \to (*,*,0)$$

where $*$ is a special symbol. As soon as the vector consisting of $n-1$ symbols $*$ followed by 0 has been constructed when t is the current position, the decoder sets $\tau = t - n + 1$ and stops.

- Given a q–ary sequence $\mathbf{Y}$, denote

$$y_i \triangleq \begin{cases} 0, & \text{if } Y_i \in \{0, \ldots, q/2 - 1\} \\ 1, & \text{if } Y_i \in \{q/2, \ldots, q-1\} \end{cases}$$

  for all $i = 0, 1, \ldots$

- Decode τ by the $(0,0,1) \to (*,*,0)$ procedure. Set $Z_i = Y_{\tau+n-1-i}$ for all $i = 0, \ldots, n-1$.

Table 2: The $(0,0,1) \to (*,*,0)$ procedure for $n = 7, \tau = 6$, and $\mathbf{y} = 1100110001011\ldots$

$\mathbf{y}$												
1	1	0	0	1	1	0	0	0	1	0	1	1
1	1	$*$	$*$	0	1	0	0	0	1	0	1	1
1	1	$*$	$*$	0	1	0	$*$	$*$	0	0	1	1
1	1	$*$	$*$	0	1	0	$*$	$*$	$*$	$*$	0	1
1	1	$*$	$*$	0	1	$*$	$*$	$*$	$*$	$*$	$*$	0

- Decode k from the standard q–ary representation $(Y_{\tau+n}, \ldots, Y_{\tau+n+l-1})$ and j from the standard q–ary representation $(Y_{\tau+n+l}, \ldots, Y_{\tau+n+2l_1-1})$, where

$$l = \lceil \log_q n \rceil, \;\; l_1 = \lceil \log_q(n+1) \rceil.$$

- Set

$$X_i' \triangleq \begin{cases} Z_{i+k}, & \text{if } i \in \{0, \ldots, n-k-1\} \\ Z_{i+k-n}, & \text{if } i \in \{n-k, \ldots, n-1\} \end{cases}$$

  and

$$X_i \triangleq \begin{cases} q-1-X_i', & \text{if } i \in \{0, \ldots, j-1\} \\ X_i', & \text{if } i \in \{j, \ldots, n-1\} \end{cases}$$

  The vector $(X_0, \ldots, X_{n-1})$ is the decoded message.

4 Encoding and decoding algorithms for the case when q is odd

4.1 Encoding algorithm

- Given a vector $\mathbf{X} \in \mathcal{Q}^n$, denote

$$x_i \triangleq \begin{cases} 0, & \text{if } X_i \in \{0, \ldots, (q-1)/2 - 1\} \\ 1, & \text{if } X_i = (q-1)/2 \\ 2, & \text{if } X_i \in \{(q+1)/2, \ldots, q\} \end{cases}$$

  for all $i = 0, \ldots, n-1$.

- Set

$$x_n \triangleq \begin{cases} 0, & \text{if } n - n_1 \text{ is even} \\ 1, & \text{if } n - n_1 \text{ is odd} \end{cases}$$

  where

$$n_1 \triangleq \left| \left\{ i \in \{0, \ldots, n-1\} : x_i = 1 \right\} \right|$$

  denotes the number of 1's in the vector $\mathbf{x}$, and set

$$X_n \triangleq \begin{cases} 0, & \text{if } x_n = 0 \\ q-1, & \text{if } x_n = 1 \end{cases}$$

– Find $j \in \{0, \ldots, n+1\}$ such that the substitutions $0 \to 2$ and $2 \to 0$ at the first $j-1$ components of the vector $(x_0, \ldots, x_n)$ transform it to the vector $(x'_0, \ldots, x'_n)$ such that

$$(x'_0 - 1) + \ldots + (x'_n - 1) = (n-1)/2$$

where

$$x'_i \triangleq \begin{cases} 2, & \text{if } x_i = 0 \text{ and } i \in \{0, \ldots, j-1\} \\ 0, & \text{if } x_i = 2 \text{ and } i \in \{0, \ldots, j-1\} \\ x_i, & \text{if } x_i = 1 \text{ or } i \in \{j, \ldots, n\}. \end{cases}$$

– Find $k \in \{0, \ldots, n\}$ such that the cyclic shift of the vector $(x'_0, \ldots, x'_n)$ by k positions to the right transforms it to the vector $(z_0, \ldots, z_n)$ having the dominating property, i.e.,

$$(z_0 - 1) + \ldots + (z_{t-1} - 1) \geq t/2$$

for all $t = 1, \ldots, n$, where

$$z_i \triangleq \begin{cases} x'_{n+i-k}, & \text{if } i \in \{0, \ldots, k-1\} \\ x'_{i-k}, & \text{if } i \in \{k, \ldots, n\}. \end{cases}$$

– Denote

$$X'_i \triangleq \begin{cases} q - 1 - X_i, & \text{if } i \in \{0, \ldots, j-1\} \\ X_i, & \text{if } i \in \{j, \ldots, n\} \end{cases}$$

and

$$Z_i \triangleq \begin{cases} X'_{n+i-k}, & \text{if } i \in \{0, \ldots, k-1\} \\ X'_{i-k}, & \text{if } i \in \{k, \ldots, n\}. \end{cases}$$

– Send the vector $(Z_n, \ldots, Z_0)$ to the channel.

– Form the standard representation of the integer k with $\lceil \log_q(n+1) \rceil$ q–ary symbols and send it to the channel.

– Form the standard representation of the integer j with $\lceil \log_q(n+2) \rceil$ q–ary symbols and send it to the channel.

4.2 Decoding algorithm

For a given ternary sequence $(y_0, y_1, \ldots)$, we will use a generalized version of the $(0,0,1) \to (*,*,0)$ presented below (example is given in Table 3).

1. Set $t = 1$.

2. If $y_t = d \neq 0$ and there are d indices $t_1, \ldots, t_d$ such that $t > t_1 > \ldots > t_d \geq 0$ with $y_{t_1} = \ldots = y_{t_d} = 0$ and $y_j = *$ for all $j \in \{t_d, \ldots, t-1\} \backslash \{t_1, \ldots, t_d\}$, then set $y_t = 0$ and $y_{t_1} = \ldots = y_{t_d} = *$.

3. If $y_t = 0$ and $y_{t-j} = *$ for $j = 1, \ldots, n$, then set $\tau = t - n$ and go to 5.

Table 3: Example of a generalized version of the $(0,0,1) \to (*,*,0)$ procedure for $q = 3$, $n = 4, \tau = 3$, and $\mathbf{y} = 00201012\ldots$

y							
0	0	2					
$*$	$*$	0	0	1			
$*$	$*$	0	$*$	0	0	1	
$*$	$*$	0	$*$	0	$*$	0	2
$*$	$*$	0	$*$	$*$	$*$	$*$	0

4. Increase t by 1 and go to 2.

5. End.

– Given a q–ary sequence $\mathbf{Y}$, denote

$$y_i \triangleq \begin{cases} 0, & \text{if } Y_i \in \{0, \ldots, (q-1)/2 - 1\} \\ 1, & \text{if } Y_i = (q-1)/2 \\ 2, & \text{if } Y_i \in \{(q+1)/2, \ldots, q\}. \end{cases}$$

– Decode τ by a generalized version of the $(0,0,1) \to (*,*,0)$ procedure. Set $Z_i = Y_{\tau+n-1-i}$ for all $i = 0, \ldots, n$.

– Decode k from the standard q–ary representation $(Y_{\tau+n+1}, \ldots, Y_{\tau+n+l})$ and j from the standard q–ary representation $(Y_{\tau+n+l+1}, \ldots, Y_{\tau+n+2l_1})$, where

$$l = \lceil \log_q(n+1) \rceil, \;\; l_1 = \lceil \log_q(n+2) \rceil.$$

– Set

$$X'_i \triangleq \begin{cases} Z_{i+k}, & \text{if } i \in \{0, \ldots, n-k\} \\ Z_{i+k-n}, & \text{if } i \in \{n-k, \ldots, n\} \end{cases}$$

and

$$X_i \triangleq \begin{cases} q - 1 - X'_i, & \text{if } i \in \{0, \ldots, j-1\} \\ X'_i, & \text{if } i \in \{j, \ldots, n\}. \end{cases}$$

The first n symbols of the vector $(X_0, \ldots, X_n)$ form the decoded message.

5 Basic ideas

We notice that vectors having the dominating property, i.e., preorder codewords for trees [6], can be easily decoded in asynchronous mode. This fact is especially helpful if we deal with binary or ternary vectors since the number of such vectors exponentially coincides with the total number of vectors. Thus, we project the given q–ary alphabet to $\{0,1\}$ or $\{0,1,2\}$ in a certain way and impelement the transformation of the initial vector to create the preorder codeword in its projection.

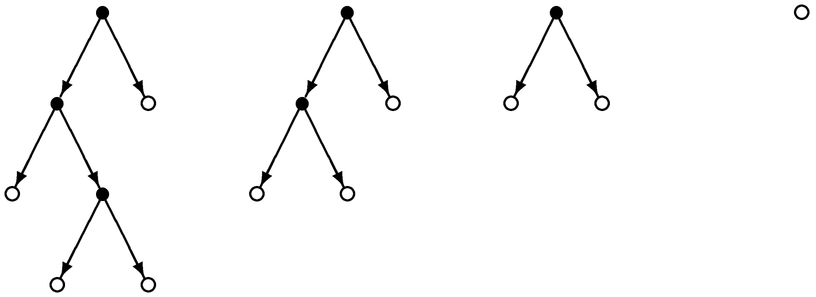

Figure 1: Transformations of the preorder codeword 1101000 of length $n = 7$ and corresponding regular binary trees.

5.1 Unique decoding of the vectors having the dominating property in asynchronous mode

Definition.

- *For any odd integer n, let the code $\mathcal{C}_n$ consist of preorder codewords for all regular binary trees having n nodes, which are taken in the reverse order, i.e., if* $\mathbf{c} = (c_0, \ldots, c_{n-1}) \in \{0,1\}^n$ *is a preorder codeword, then* $\overline{\mathbf{c}} \triangleq (c_{n-1}, \ldots, c_0) \in \mathcal{C}_n$.
- *Given integer n, let the code $\tilde{\mathcal{C}}_n$ consist of ternary preorder codewords for all non–regular binary trees having n nodes, which are taken in the reverse order.*

For example,

$$\mathcal{C}_5 = \Big\{ 00011, 00101 \Big\}$$
$$\tilde{\mathcal{C}}_5 = \Big\{ 00112, 01102, 01012, 00121, 01021, 00211, 00022, 00202, 01111 \Big\}.$$

Proposition. *For any $\tau \in \{0, \ldots, n-1\}$, any of transmitted codewords $\overline{\mathbf{c}} \in \mathcal{C}_n$ or $\overline{\mathbf{c}} \in \tilde{\mathcal{C}}_n$ can be uniquely decoded if $(y_\tau, \ldots, y_{\tau+n-1}) = \overline{\mathbf{c}}$ and $y_0, \ldots, y_{\tau-1}$ are arbitrary binary or ternary symbols.*

Proof : Let us consider the case when the code $\mathcal{C}_n$ is used for data transmission. The decoder can represent the received binary sequence $(y_0, y_1, \ldots)$ as the path on the plane going through the points (t, σ'_t), where $\sigma'_t \triangleq 2(y_0 + \ldots + y_{t-1}) - t$ and $t = 1, 2, \ldots$ His goal is to find the position of the axes (τ, σ'_τ), where $\tau \in \{0, \ldots, n-1\}$, in such a way that the first n steps of the path have the dominating property in the reverse order, i.e., $\sigma'_i < \sigma'_\tau$ for all $i = \tau+1, \ldots, \tau+n-1$ and $\sigma'_{\tau+n-1} = \sigma'_\tau - 1$. If a codeword of the code $\mathcal{C}_n$ was transmitted, then a solution to this problem exists, and we only have to check that this solution is unique.

Suppose that $\sigma'_i < \sigma'_\xi$ for all $i = \xi+1, \ldots, \xi+n-1$ and $\sigma'_{\xi+n-1} = \sigma'_\xi - 1$ for some $\xi \in \{0, \ldots, n-1\} \backslash \{\tau\}$. If $\xi > \tau$, then $\sigma'_\xi < \sigma'_\xi$ and $\sigma'_{\tau+n-1} = \sigma'_\xi - 1$ contradicts $\sigma'_{\tau+n-1} < \sigma'_\xi$. If $\xi < \tau$ and $\sigma'_\xi \geq \sigma_\xi$, then $\sigma'_{\xi+n-1} = \sigma'_\xi - 1$ contradicts $\sigma'_{\xi+n-1} < \sigma'_\xi$. It is easy to see that the case $\xi < \tau$ and $\sigma'_\xi < \sigma'_\xi$ is also not possible.

The proof for the case $\overline{\mathbf{c}} \in \tilde{\mathcal{C}}_n$ is similar : we define $\sigma'_t \triangleq y_0 + \ldots + y_{t-1} - t$ for all $t = 1, 2, \ldots$ and proceed as before.

The $(0,0,1) \to (*,*,0)$ procedure and its generalized version included in the decription of decoding algorithms are derived from the algorithm that can be viewed as sequential cuts of leaves of the tree (see Figure 1, where the $*$ symbols can be considered as deletions).

5.2 Transformations $\mathbf{x} \to \mathbf{x}' \to \mathbf{z}$

Let us consider the case when q is even and n is odd. The transformation $\mathbf{x} \to \mathbf{x}'$ is based on the following fact discovered by D. E. Knuth [7]. Let $\mathbf{x}_0 \triangleq \mathbf{x}$ and

$$\mathbf{x}_i \triangleq (x_0 \oplus 1, \ldots, x_{i-1} \oplus 1, x_i, \ldots, x_{n-1})$$

for all $i = 1, \ldots, n$. Let w_i denote the number of ones in the vector $\mathbf{x}_i$ (the Hamming weight of $\mathbf{x}_i$) for all $i = 0, \ldots, n$. Then the path on the plane, specified by the points $(0, w_0), \ldots, (n, w_n)$, has increments ± 1 and there always exists a $j \in \{0, \ldots, n\}$ such that $w_j = (n-1)/2$.

Let $s_0 \triangleq 0$ and let

$$s_t \triangleq 2(x_0 + \ldots + x_{t-1}) - t, \text{ for all } t = 1, \ldots, n$$

denote the difference between the number of ones and the number of zeroes in the first t positions of the vector $\mathbf{x}_0$. Then

$$x_0 + \ldots + x_{j-1} = (s_j + j)/2$$

and

$$x_j + \ldots + x_{n-1} = (s_n - s_j + n - j)/2.$$

Therefore, since

$$(x_0 \oplus 1) + \ldots + (x_{j-1} \oplus 1) = j - (x_0 + \ldots + x_{j-1})$$

the requirement that $\mathbf{x}_j$ is a balanced vector is equivalent to the equation

$$s_n = 2s_j - 1. \tag{3}$$

We define j as the smallest index, for which (3) holds.

The transformation $\mathbf{x}' \to \mathbf{z}$ is based on the following statement, known as the Raney's lemma [8] : *for any odd n, the set of all binary vectors of length n containing $(n+1)/2$ zeroes and $(n-1)/2$ ones can be uniquely partitioned in*

$$\frac{1}{n}\binom{n}{(n-1)/2}$$

cyclic groups, and each group contains exactly one vector having the dominating property. Furthermore, if this vector has to be received by the cyclic shift of a given balanced vector $\mathbf{x}'$ by k positions to the right, then the value of the parameter k is defined by

$$\begin{cases} s'_i > s'_k, & \text{if } i \in \{0, \ldots, k-1\} \\ s'_i \geq s'_k, & \text{if } i \in \{k, \ldots, n-1\} \end{cases} \tag{4}$$

where $s'_0 \triangleq 0$ and $s'_t \triangleq 2(x'_0 + \ldots + x'_{t-1}) - t$ for all $t = 1, \ldots, n$.

It is easy to see that

$$s'_i = \begin{cases} -s_i, & \text{if } i \in \{0, \ldots, j-1\} \\ s_i - 2s_j, & \text{if } i \in \{j, \ldots, n-1\}. \end{cases}$$

Thus, the knowledge of the sequence $s_1, \ldots, s_n$ is sufficient for organizing the transformations $\mathbf{x} \to \mathbf{x}' \to \mathbf{z}$.

If q is odd, then generalization of the approaches above is straightforward : we attach "a parity ternary symbol" to the vector $(x_0, \ldots, x_n)$, since trees described by the given vector do not exist otherwise, and process the vector of length

$$n' \triangleq \begin{cases} n + 1 - n_1, & \text{if } n - n_1 \text{ is even} \\ n - n_1, & \text{if } n - n_1 \text{ is odd} \end{cases}$$

which is obtained if we puncture all 1's in the vector $(x_0, \ldots, x_n)$. Namely (see (3) and (4)), j can be defined as the smallest index, for which

$$\hat{s}_{n+1} = 2\hat{s}_j - 1$$

and k is defined by

$$\begin{cases} \hat{s}'_i > \hat{s}'_k, & \text{if } i \in \{0, \ldots, k-1\} \\ \hat{s}'_i \geq \hat{s}'_k, & \text{if } i \in \{k, \ldots, n\} \end{cases}$$

where $\hat{s}_0 = \hat{s}'_0 \triangleq 0$ and

$$\begin{aligned} \hat{s}_t &\triangleq (x_0 - 1) + \ldots + (x_{t-1} - 1) \\ \hat{s}'_t &\triangleq (x'_0 - 1) + \ldots + (x'_{t-1} - 1) \end{aligned}$$

for all $t = 1, \ldots, n+1$.

6 Conclusions

Our mathematical problem seems to be close to practical requests. We present two situations.

1). The participants are moving. The receiver wants to know the current distance and simultaneously receive messages from the sender. Therefore, the latter one splits data in pieces of length n and sends them with some fixed delay known to the receiver. The value of the variable delay τ gives information about the distance.

2). Two people communicate with each other at dialog by sending packets consisting of n bits. The time for the reaction depends on many factors and varies from packet to packet. We allow the variation by at most $n-1$ bits.

References

[1] V. B. Balakirsky (2001) A new coding algorithm for trees. *Comp. J.*

[2] V. B. Balakirsky (2001) Block codes for asynchronous data transmission designed from binary trees. *Comp. J.*

[3] V. I. Levenshtein (1964) Decoding automata that are invariant to the initial state. *Problems of Cybernetics,* **12** (in Russian), 125–136.

[4] V. I. Levenshtein (1970) On the maximum number of words in codes without overlaps. *Probl. Inform. Transmission,* **4**, 88–90.

[5] H. B. Mann (1962) On the number of information symbols in Bose–Chaudhuri codes. *Inform. Control,* **5**, 153–162.

[6] E. Mäkinen (1996) A survey on binary tree codings. *Comp. J.,* **34**, 438–443.

[7] D. E. Knuth (1986) Efficient balanced codes. *IEEE Trans. Inform. Theory,* **32**, 51–53.

[8] R. L. Graham, D. E. Knuth, and O. Patashnik (1989) *Concrete Mathematics.* New York : Addison–Wesley.