

On the principal state method for runlength limited sequences

Citation for published version (APA):

Tjalkens, T. J. (1994). On the principal state method for runlength limited sequences. *IEEE Transactions on Information Theory*, 40(3), 934-941. <https://doi.org/10.1109/18.335906>

DOI:

[10.1109/18.335906](https://doi.org/10.1109/18.335906)

Document status and date:

Published: 01/01/1994

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

ON THE PRINCIPAL STATE METHOD FOR RUNLENGTH LIMITED SEQUENCES.

Tjalling Tjalkens
Eindhoven University of Technology,
P.O.Box 513, 5600 MB Eindhoven,
The Netherlands.

Abstract.

We present a detailed result on Franaszek's principal state method for the generation of runlength constrained codes. We show that, whenever the constraints k and d satisfy $k \geq 2d > 1$, the set of "principal states" is s_0, s_1, \dots, s_{k-1} . Thus there is no need for Franaszek's search algorithm anymore. The counting technique used to obtain this result also shows that "state independent decoding" can be achieved using not more than three codewords per message and it allows us to compare the principal state method with other practical schemes originating from the work of Tang and Bahl and also allows us to use an efficient enumerative coding implementation of the encoder and decoder.

Introduction.

Shannon [4] considers the (d, k) -constrained channel, where the only possible binary sequences that can be transmitted over the channel are those containing runs of zeros of length d, \dots, k . ($d < k$). These channels can be described by a state model where each state is indexed by the length of the current run of zeros. Shannon defines the capacity of this channel as the limit as $n \rightarrow \infty$ of the logarithm of the size of the set of all sequences of length n satisfying the (d, k) -constraint divided by n .

A runlength constrained code, (d, k) -constrained code, is a binary encoding of information such that in the code sequence successive ones are separated by at least d zeros and at most k zeros and thus is well suited for use on a (d, k) -constrained channel.

We shall consider fixed length codes for these purposes. Valid codewords follow a possible path in this state model, starting at the state where the previous codeword ended. So, a code for this state model contains several codewords sets, each containing a variable number of words, where the selected set depends on the previous codeword and is such that the concatenation of that codeword with any word in the set is permissible. Since we consider fixed length codes the size of the code is determined by the smallest set belonging to some state in the model.

Franaszek [3] noted that if we take a subset of all states in the model and require the codewords to start and end in states of this subset then an optimum subset exists. This subset is known as the set of principal states and Franaszek described an algorithm to search for these principal states.

Another approach, presented by several authors, [1, 5, 6], is to use a single set S of codewords that satisfy the (d, k) -constraint internally. A special sequence is put in between two codewords such that the (d, k) -constraint remains satisfied between codewords.

The principal state method is an optimal code for systems that can be described in the state model framework, and thus it is at least as efficient as any of the glue methods, since the glue methods can also be described in the state model framework.

The principal states.

Our goal

We start with the definition of the building blocks or basic sets $U(m)$ for the codeword sets given the (d, k) -constraint, containing all sequences that start and end with a "one" and satisfy the (d, k) -constraint internally. Let $|U(m)|$ denote the size of $U(m)$.

In the following we shall repeatedly use the shorthand notation $[a; b] \triangleq \{a, a+1, \dots, b\}$.

Let $S \subset [0; k]$ denote the set of permitted channel states, (not necessarily the set of principal states). Consider the sets $V_S(n; i)$ containing all sequences starting with a run of i zeros and ending in a run of $r \in S$ zeros and satisfying the (d, k) -constraint internally. Note that $V_S(n; i)$ can be described using the basic sets as

$$V_S(n; i) = \bigcup_{j \in S} \{0^i\} * U(n-i-j) * \{0^j\},$$

where $U * V$ indicates the set containing all concatenations of the sequences $\underline{x} \in U$ with any sequence $\underline{y} \in V$.

With these sets we can make Franaszek's state depending codeword sets $W_S(n; i)$, i.e. the set of possible codewords of length n starting in state $i \in S$ and ending in any state $j \in S$. We have

$$W_S(n; i) = \bigcup_{\max\{0, d-i\} \leq j \leq \min\{n, k-i\}} V_S(n; j).$$

Now we can formulate our goal and the result:

Given n, k , and d , (with the restriction $n \geq k \geq 2d > 0$), find the set $S^* \subset [0; k]$ such that:

$$W_{S^*}(n) \triangleq \max_{S \subset [0; k]} \min_{i \in S} |W_S(n; i)| = \sum_{i=0}^{k-1} \sum_{j=0}^{k-d} |U(n-d-i-j)|.$$

Message mapping for state independent decoding

Partition the W_{S^*} messages into sets M_i of sizes $|M_i| \triangleq |V_{S^*}(n; d+i)|$, where $i = 0, 1, \dots, k-d$. Let r be the number of trailing zeros in the previous codeword. We distinguish between the following cases:

$d = 1$ and $r = 0$: The messages in the set M_i are assigned to the set $V_{S^*}(n; i+1)$.

$d = 1$ and $r \geq 1$: The set M_0 is assigned to $V_{S^*}(n; 1)$ and $M_1 \cup \dots \cup M_{k-1}$ are assigned to $V_{S^*}(n; 0)$.

$d > 1$ and $r < d$: For all $i = 0, \dots, k-d-r$ we assign to the set M_i the codewords from $V_{S^*}(n; d+i)$ respectively. For $i = k-d-r+1, \dots, k-d$ we assign to the set M_i the codewords from $V_{S^*}(n; i+2d-k+1)$ respectively.

$d > 1$ and $r \geq d$: The sets $M_0 \cup M_1 \cup \dots \cup M_{k-d}$ are assigned to $V_{S^*}(n; 0)$ and $V_{S^*}(n; 1)$ in that order.

So, it is easy to see that every message is encoded into one of two or three different codewords, depending on r .

Enumerative coding.

We shall briefly indicate the application of the well-known enumerative coding technique [2] to the generation of the (d, k) -constrained sequences.

First we determine the message subset M_i of the message m that we want to transmit. Then, with the rules of the previous section we determine the set $V_{S^*}(n; j)$ and the relative index $i(\underline{x}^n; V_{S^*}(n; j))$ of our message in the set. Finally we use the enumerative reconstruction to produce the codeword $\underline{x}^n \in V_{S^*}(n; j)$ from its index.

Let the codeword \underline{x}^n be given as $\underline{x}^n = 0^{\alpha_0} 10^{\alpha_1} 1 \dots 10^{\alpha_p}$. So $\alpha_0 = j$.

Although we will not need the (source) encoding algorithm, it is instructive to see how the index can be computed recursively as

$$\begin{aligned} i(\underline{x}^n; V_{S^*}(n; j)) &= i(\underline{x}_{j+1}^n; V_{S^*}(n-j; 0)) = \\ &= i(\underline{x}_{\alpha(\underline{x}^n)+2}^n; V_{S^*}(n-\alpha(\underline{x}^n)-1; 0)) + \sum_{l=d}^{\alpha(\underline{x}^n)-1} |V_{S^*}(n-l-1; 0)|, \end{aligned}$$

where $\alpha(\underline{x}^n) = \alpha_1$ as given above.

Note that this computation produces a lexicographical ordering given the symbol ordering " $1 < 0$ ". Also note that in order to compute the index we only need the $n+1$ numbers $|V_{S^*}(p; 0)|$ for $0 \leq p \leq n$.

Reconstructing \underline{x}^n involves producing the $\alpha_0, \dots, \alpha_p$ and they can be found recursively by the corresponding enumerative decoding algorithm.

References.

- [1] C.F.M. Beenker and K.A. Schouhamer Immink, "A generalized method for encoding and decoding run-length-limited binary sequences," *IEEE Trans. Inform. Theory*, vol IT-29, pp. 751-754, Sept. 1983.
- [2] T.M. Cover, "Enumerative source encoding," *IEEE Trans. Inform. Theory*, vol IT-19, pp. 73-77, Jan. 1973.
- [3] P.A. Franaszek, "Sequence-state coding for digital transmission," *B.S.T.J.*, vol 47, pp 143-157, Jan. 1968.
- [4] C.E. Shannon, "A mathematical theory of communication," *B.S.T.J.* vol 27, pp. 379-423, July 1948.
- [5] D.T. Tang and L.R. Bahl, "Block codes for a class of constrained noiseless channels," *Information and Control*, vol 17, pp. 436-461, 1970.
- [6] J.H. Weber and K.A.S. Abdel-Ghaffar, "Methods for cascading runlength-limited sequences," *Proc. Twelfth Symposium on Information Theory in the Benelux*, Veldhoven, May 23 & 24, 1991.