TECHNICAL TRANSACTIONS 10/2018
CZASOPISMO TECHNICZNE 10/2018

ELECTRICAL ENGINEERING

DOI: 10.4467/2353737XCT.18.150.9098
SUBMISSION OF THE FINAL VERSION: 28/09/2018

Denys Gutenko orcid.org/0000-0003-4160-724X
dgutenko@pk.edu.pl
Department of Automation and Information Technologies, Faculty of Electrical and Computer Engineering, Cracow University of Technology

Selecting quasi-constant weight code parameters for systems of automatics

Wybór parametrów kodu quasi-równoważnego w systemach automatyki

Abstract

In this paper, the requirements of quasi-constant weight code parameters are considered for a certain number of encoded messages and for various requirements of noise immunity. An algorithm is proposed for choosing the optimal parameters of a quasi-constant weight code for a given number of encoded messages and a level of noise immunity that would ensure maximum speed of control command transmission.

Keywords: quasi-constant weight codes, communication channel, noise immunity, speed of control command transmission

Streszczenie

W artykule uwzględniono wymagania dotyczące parametrów kodu quasi-równoważnego w odniesieniu do liczby zakodowanych komunikatów dla różnych wymagań w zakresie odporności na szum. Został zaproponowany algorytm wyboru optymalnych parametrów kodu quasi-równoważnego dla podanych liczby zakodowanych komunikatów i poziomu odporności na szum, który zapewniłby maksymalną prędkość przesyłu komunikatów.

Słowa kluczowe: kody quasi-równoważne, kanał komunikacyjny, odporność na hałas, prędkość przesyłu komunikatów.

Symbols

N, K	–	numbers of encoded messages (the number of control signals)
n	–	number of digits in a combination
k	–	number of units in a combination
α	–	half the width of the interval of the allowed values of the number of units in the combination
β	–	half the width of the segment of unacceptable values of the number of units in the combination
D, P	–	fractions of detectable errors

1. Introduction

In modern automation systems, the actual task remains to increase the speed of transmitted control signals with the required noise immunity. In control systems, communication channels are often asymmetric. For asymmetric channels, it is advisable to select constant weight codes since with an increase in the degree of asymmetry of the channel, the detectability of these codes increases. In an absolutely asymmetric communication channel, constant weight codes allow 100% of errors to be detected. In addition, for a given code word length, the chosen constant weight code usually has more resolved combinations than the separable code with the same detecting ability [1]. Another advantage of constant weight codes is the ease of detecting errors by counting the number of units in a combination. As a consequence, coding and decoding devices are simple; this is why these codes are widely used. For example, one of these codes is a five-digit code with two units and a seven-digit code with three units [2].

The main drawback of constant weight codes is their inseparability; therefore, in the code combination, it is necessary to convert input words into constant weight combinations. [1]. In addition, constant weight codes cannot detect errors when one or more units tend to zero at the same time and the same number of zeros tend to units in the transposed combination [2]. Additionally, constant weight codes should be selected for communication channels with a certain level of noise immunity. However, in some control systems over time, the requirements for noise immunity of the communication channel may change. This is why, in such cases, it would be useful to change the level of noise immunity of transmitted combinations, so we could obtain a gain in speed in cases where high noise immunity is not needed. In this paper, speed will be understood as the number of control commands sent per unit of time. To increase speed, it is necessary to encode a control command using a combination with the fewest number of bits since in this case, it would be possible to send less bits through communication channel for one control command. In standard constant weight code, it is impossible to change the length of code combination. A code in which the length of the code combinations changes with noise immunity is called the quasi-constant weight (code with quasi-constant weight) [3, 4, 5]. Therefore, the task emerges of developing an algorithm for selecting a quasi-constant weight code which provides maximum speed with changing requirements for noise immunity.

2. Choice of quasi-constant weight code

Let us suppose that there is a control system with a communication channel through which K different control commands can be transmitted. It is necessary to choose a quasi-constant weight code that could be used to transfer command data, satisfy the changing noise immunity requirements and at the same time, can provide the best possible speed for these requirements.

If the requirements for noise immunity are unchanged, the quasi-constant weight code becomes a normal constant weight code. This means that the quasi-constant weight code can be represented as a constant weight code on time intervals with unchanged requirements for noise immunity. This is why the operating time of the control device can be divided into a certain number of time intervals on which the requirements for noise immunity are constant. Thus, at each of these intervals, combinations of constant weight codes corresponding to the given noise immunity are sent. The parameters of the constant weight codes depend on the noise immunity requirements for the corresponding interval. In addition, the number of constant weight combinations must be greater than or equal to the number of control commands K:

$$N_p \geq K, \tag{1}$$

where N_p is the number of combinations.

In general, the number of combinations of the constant weight code is [6]

$$N_p = \frac{n!}{(n-k)!k!} \tag{2}$$

where n is the number of digits of the combination and k is the number of units.

By substituting (2) into (1), a new formula is obtained:

$$\frac{n!}{(n-k)!k!} \geq K \tag{3}$$

Therefore, for each constant weight code used in a quasi-constant weight code, the parameters n and k must have values with which inequality (3) is satisfied. From inequality (3), it is possible to get parameters n and k by analysing the corresponding graphs or by the method of selection. Figure 1 presents variants of the graphical solution of the inequality (3) for different n and k with the number of control signals $K = 35$.

In Fig. 1, continuous curves show the relationship between the number of allowed combinations and different k with some fixed parameter n for each curve. By the nature of the changes in this figure, one can see that the number of allowed combinations reaches the maximum value for $k = n/2$ and gradually decreases from this value to the minimum value for $k = 0$ and $k = n$. Since n and k can only take discrete values, it should be understood that the relationship between the number of allowed combinations and the parameter k is a collection of points that are connected in a continuous curve for easy perception.

The solution of inequality (3) is the collection of curve points that are on or above the line $K = 35$. As can be seen in Fig. 1, there is a minimum value of $n = n_{min}$ for which there are

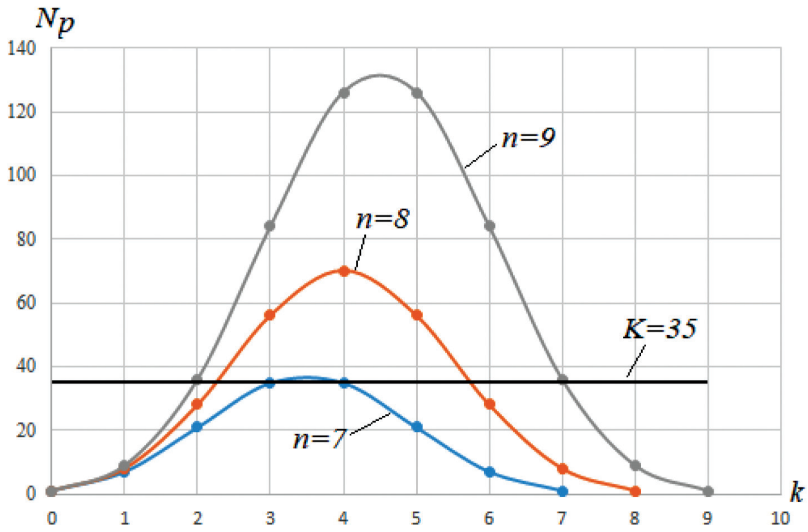

Fig. 1. Options for solving inequality (3) for different n and k for $K = 35$

solutions of this inequality. As n increases, the number of possible solutions increases. For $n < n_{min}$, the inequality has no solutions since there are no integer nonnegative k for which it is satisfied.

For a fixed n, the range of values of k at which the corresponding curve is above the line $K = const$ is symmetrical with the centre of symmetry at $n/2$. This means that the range of values of k at which inequality (3) is satisfied in the general case can be written in the form $k \in \left[\frac{n}{2} - \alpha_i ; \frac{n}{2} + \alpha_i\right]$, where α is a parameter that determines the width of the range of values of k for a certain $n = n_i$ if solutions of inequality (3) exist. As can be seen in Fig. 1, with increasing n value, the range of solutions of k also increases, so for $n_j > n_i$, $\alpha_j > \alpha_i$.

For the considered example, the minimum value of n for which inequality (3) would be satisfied is $n_{min} = 7$. The range of solutions k for a given n is in the range 3 to 4 and is symmetrical with the centre of symmetry at $n/2 = 3.5$. However, k cannot take such a value, since it must always be integer. In this interval, there are two non-negative integer values $k = 3, 4$ which are the solution of this inequality.

For $n = 8$, the range of solutions of k is symmetrical with the centre at $n/2 = 4$, and there are already three integer values $k = 3, 4, 5$ on this range, which are the solutions for this inequality. For $n = 9$, the range of solutions of k is symmetrical with the centre at $n/2 = 4.5$ and on this range, there are already six integer values k = 2, 3, 4, 5, 6, 7, which represent solutions for inequality (3). With increasing n, the number of possible solutions for k increase further.

It should be understood that despite increased numbers of decision for inequality, with increased length n of the combination the speed will be also reduced, because more bits must be transmitted for each control command. Therefore, with imposed noise immunity requirements and a certain number of control commands K, it is necessary to select a constant weight code with the minimum number of bits in combinations.

To estimate the noise immunity, the formula [1] of the fraction of detectable errors is used:

$$D = 1 - \frac{N_p}{N_{\mathbf{B}}}, \tag{4}$$

where N_B is the number of all possible combinations.

The fraction of detectable errors is characteristic of the code itself and is not characteristic of the communication channel. Since the number of bits of the constant weight code is described by the parameter n, the number of all possible combinations N_B with this number of bits is equal to the number of combinations of the binary natural code 2^n. By substituting formula (2) into (4), the fraction of detectable errors for the constant weight code with n-bit combinations having k units is obtained:

$$D = 1 - \frac{\frac{n!}{(n-k)!k!}}{2^n}. \tag{5}$$

Let us assume that for the communication channel in the control system, there are requirements for the noise immunity level that can be expressed as the required fraction of detectable errors in the used code. Such a required fraction is described with the parameter P. As mentioned earlier, the operating time of the control system was divided into intervals in which the requirements for noise immunity are unchanged. This means that for each of these intervals $P = const$, and for each of them, it is necessary to choose a constant weight code which satisfies the requirements of its noise immunity. This is why it is necessary to choose a constant weight code with parameters n and k for which the fraction of detectable errors (5) should be greater than or equal to the parameter P.

$$1 - \frac{\frac{n!}{(n-k)!k!}}{2^n} \geq P. \tag{6}$$

Inequality (6) can be solved either graphically or by the method of selection. Figure 2 shows the solutions for different n and k for the parameter $P = 0.85$.

Figure 2 shows the curves, each of which is the relationship between the fraction of detectable errors and the number of units k in constant weight combinations with fixed number of bits n. Since k takes only discrete values, this relationship can be expressed by a collection of points that are connected in continuous curves for easy perception of the figure.

The solution of inequality (6) is the collection of curve points that are above the line $P = 0.85$. As can be seen in the figure, in the general case for a fixed n, the solution of this inequality for parameter k is two intervals. The beginning of the first interval is always at 0 and the end of the second interval is always at n. The end of the first interval and the beginning of the second interval are symmetrically located with the centre of symmetry at point $n/2$. This is why, in the general case, the solution of inequality (6) for some $n = n_i$ can be expressed

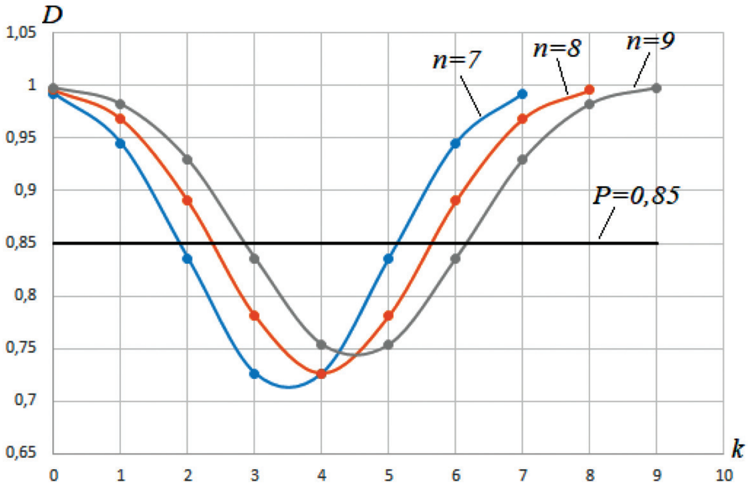

Fig. 2. Options for solving inequality (6) for different n and k for $P = 0.85$

as $k \in \left[0; \frac{n}{2} - \beta_i\right] \cup \left[\frac{n}{2} + \beta_i; n\right]$, where $\beta_i \geq 0$ is the parameter that describes the width of the interval at which inequality (6) has no solutions. As can be seen in Fig. 2, if n increases, the number of possible solutions for k also increases.

For this example, for $n = 7$, the range of possible solutions of k is at intervals [0, 1] and [6; 7]. The end of the first interval and the beginning of the third interval are symmetrically located with a centre of symmetry at point $n/2 = 3.5$. Thus, solutions for inequalities (3) and (6) were obtained. The next step is to find a general solution for both inequalities (3) and (6) since such a general solution would correspond to the parameters of a quasi-constant weight code which satisfy all requirements. Such a solution is the intersection of two intervals obtained from the inequalities (3) and (6). Figure 3 a) shows the intersection of these intervals if $\beta < \alpha$. The result of this intersection is two intervals $\left(\left[\frac{n}{2} - \alpha; \frac{n}{2} - \beta\right] \text{ and } \left[\frac{n}{2} + \beta; \frac{n}{2} + \alpha\right]\right)$ of k values. At these intervals, it is necessary to choose the optimal k. Since the choice of k is made with fixed n, it will not affect the speed of the system. This is why the criterion of choice should be something else; for example, it could be additional noise immunity. Despite the fact that all values of k satisfy inequality (6), and thus the requirements for noise immunity, additional noise immunity is not superfluous, considering that other parameters do not suffer at the same time. This is why at these intervals, you can choose the best k for the noise immunity parameters. According to Fig. 2, the further k is from $n/2$, the higher the noise immunity is. Since $\beta < \alpha$, the most distant from $n/2$ are the points $\frac{n}{2} - \alpha$ and $\frac{n}{2} + \alpha$.

If $\beta > \alpha$, there is no intersection between intervals, and the general solution for inequalities (3) and (6) is an empty set (Fig. 3b). If $\beta = \alpha$, then the solution is only two points $\frac{n}{2} - \alpha = \frac{n}{2} - \beta$ and $\frac{n}{2} + \alpha = \frac{n}{2} + \beta$.

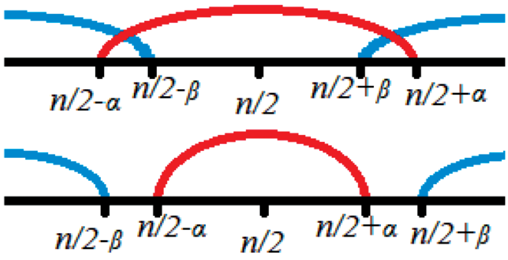

Fig. 3. Variants of general solution for inequalities (3) and (6)

By using the obtained results, it is possible to receive a general algorithm for selecting parameters *n* and *k* of the quasi-constant weight code at each time interval with a constant requirement for noise immunity determined by the permissible fraction of the detected error *P* and a given number of control commands *K* in order to obtain the maximum speed.

1. Set $n = 2$.
2. Set $k = n/2$ if *n* is even. If *n* is not even than set *k* to the closest integer value, which is less than $n/2$.
3. Check whether the inequality (3) holds with such parameters *n* and *k*. If inequality (3) doesn't hold, increase *n* by one and go to p. 2. If inequality (3) holds, go to p. 4.
4. Decrease *k* by one and check whether the inequality (3) holds with such parameters *n* and *k*. If inequality (3) doesn't hold, increase *k* by one and go to p. 5. If inequality (3) still holds, go to p. 4
5. Check whether the inequality (6) holds with such parameters *n* and *k*. If inequality (3) doesn't hold, increase *n* by one and go to p. 2. If inequality (3) still holds, go to p. 6.
6. Select current parameters *n* and *k* as parameters of quasi-constant weight code for transmission.
7. The end of the algorithm.

At the end of the obtained algorithm, the quasi-constant weight code with the minimum number of digits *n* of a combination is received, for which inequalities (3) and (6) are satisfied. This algorithm allows obtaining the maximum speed for a given number of control commands and noise immunity requirements for each time interval. Since this algorithm must be performed at each time interval with an unchanged noise immunity requirement, the number of bits change, adapting to the changing requirements of noise immunity while striving to maintain the maximum speed.

3. Conclusion

Thus, the proposed method allows a given set of control commands to select quasi-constant weight code parameters that provide the maximum speed at the required level of noise immunity. Such quasi-constant weight codes can be used for the control command transmission in systems of automatics [2].

References

[1] Berezyuk N.T., *Information encoding*, Kharkiv 1978.
[2] Tutevich V.N., *Telemechanics*, Moscow 1985.
[3] Tsymbal V.P., *Teriya information and coding*, 1992, 263.
[4] Peter Fenwick, *A Note on Variable-Length Codes with Constant Hamming Weights*, Journal of universal computer science, Vol. 21, issue 9, 2015.
[5] Kulik I.A., Skordina Ye.M., Posny S.N., *Error-detecting Ability of Quasi-Equilibrium Code of Sumy*, Bulletin of Sumy State University Series Technical Sciences, No. 1, 2012, pp. 100–111.
[6] Borisenko A.A., Berezhnaya O.V., Kulik I.A., *Estimation of the noise immunity of the data transmission system on the basis of equilibrium codes*, Sumy: Visnyk of Sumy State University, No. 1(12), 1999, pp. 79–82.

If you want to quote this article, its proper bibliographic entry is as follow: Gutenko D., *Selecting quasi-constant weight code parameters for systems of automatics*, Technical Transactions, Vol. 10/2018, pp. 119–126.