Organizing Gray Code States for Maximum Error Tolerance

Conference Paper · January 2009

2 authors:

Nicholas Harkiolakis

Lambros Ekonomou
UBITECH ENERGY Sprl

Organizing Gray Code States for Maximum Error Tolerance

NICHOLAS HARKIOLAKIS

School of Electrical and Computer Engineering
National Technical University of Athens
9 Iroon Politechniou St., 157 80 Athens
GREECE

e-mail: nharkiolakis@computer.org

LAMBROS EKONOMOU

Department of Electrical Engineering Educators
School of Pedagogical and Technological Education
N. Heraklion, 141 21 Athens
GREECE

e-mail: leekonom@gmail.com

Abstract: - An alternative to Gray codes is presented where the error between two codes is proportionally analogous to the distance of their positions. The new coding scheme (Selected coding) uses a certain subset of Gray codes of higher length that the ones that would normally being used to carry a given signal. This makes efficient coding for cyclic values that represent rotational variables. This makes it ideal for representing cyclic patterns in neural networks and genetic algorithms. Case studies are used to present the coding scheme and its error handling efficiency.

Key-Words: - Coding, rotational variables, Gray codes

1. Introduction

Binary numeral systems where successive values differ in one bit are well know and frequently used to facilitate error correction in digital communications [1-7] and in pattern representations in artificial intelligence technologies like neural networks and genetic algorithms. The most popular forms of coding used today are Gray codes and their variations. They were initially developed by Frank Gray [1] in an attempt to minimize the errors in converting analog signals to digital. The problem at hand was the cyclic nature of values that are expressed in terms of rotary variables and are always a function of an angle. In terms of angles and in particular for trigonometric functions there is a periodicity of 360 degrees. And while numerically all multiples of 360 like 0, 360, 720, etc are different, in terms of trigonometric functions and rotational variables they are exactly the same.

One needs to have a method of gradually transitioning during a cycle back to the original values and minimize as much as possible positional error in that neighboring regions should differ as little as possible (a vital property for switches that can not be perfectly synchronized). The solution comes with the reflected binary code that changes one switch at a time so there are no position errors. As can be easily seen in the simple case of Table 1 (where binary codes of three bit length are displayed), positions 3 and 4 (decimal) while next to each other in their binary representation they differ in two positions. The corresponding Gray coding though removes the problem and all neighbors differ by only one position.

This is ideal for coding were errors in one position only need to be addressed. For example gray codes of decimal positions 0, 1, 3, and 7 can be used to represent the “0” state while the rest can be used to represent the “1” state. It is obvious that the declared redundancy can easily handle one digit errors since regardless of their error position they will interpret the correct one digit states of “0” and “1”. What happens though if we have very noisy signals were errors in more than one position can occur? With Gray coding it is unlikely to have any indication as to where our correct value might be. Table 1 (leftmost columns) lists the coding produced when two and three errors occur (see respective columns). Allowing for two errors we see that only two schemes (000 and 111) produce the correct result (“0” and “1”) while the rest can not be uniquely assigned. Regarding signals with three position errors we see that the three digit Gray coding collapses completely making it impossible to uniquely identify any pattern.

Decimal Coding	Binary Coding	Gray Coding	1	2	3
			positional error		
0	000	000	0	0	0, 1
1	001	001	0	0, 1	0, 1
2	010	011	1	0, 1	0, 1
3	011	010	0	0, 1	0, 1
4	100	110	1	0, 1	0, 1
5	101	111	1	1	0, 1
6	110	101	1	0, 1	0, 1
7	111	100	0	0, 1	0, 1

Table 1. Binary and Gray code representations and their corresponding interpretations.

Another observation we can make is that there is no pattern or relationship in the diffusion of error across neighbors meaning that one can not tell if a two position error for example will result in having the correct value between neighbors at certain distance from each other. In terms of variables (rotational) that represent angles it is very important if in the presence of errors we can predict the correct value with certain accuracy.

To eliminate this inefficiency of Gray coding to accurately interpret multi error codes an alternative binary coding has been developed. The code could be seen as a variation of Gray codes since its elements can be found in Grey codes albeit in higher order (code length) than the ones used to represent a given code length. What is different now is the distribution of the binary digits that follows a predefined pattern that while complies with the Grey codes principle to have successive values differ in only one bit it has the additional property of having values differ between themselves as a linear function of the absolute difference of their positions. More specifically the function in question will be the identity function.

2. Methodology

The developed coding scheme ensures that consecutive codes differ by one bit and if we have two values at positions ***i*** and ***j*** separated by a distance **|i-j|** the difference ***e*** in their bit values will be |i-j|. So ***e*** **=|** ***j-i*** **|.** To demonstrate the new coding scheme **t**he developed codes are shown in Table 2 as "Selected Coding" (meaning selected values from Gray codes) along with the equivalent Gray codes of length 4. In addition the interpretation according to the number of errors is listed.

As we can see from the above table and for the case that we want to make the correct binary interpretation while for one error the Gray codes more reliably interpret the pattern for more errors the developed codes seem to behave better. To follow one example let as consider the case of the third code row of the table. Our intention here is to interpret the correct value for 0 and 1. Using Gray codes and allowing for one bit errors we have 0000, 0001, 0010, 0100, 1000, all reliably interpreted as "0" while 0111, 1011, 1110, 1111, 1101 reliably interpreted as "1". The rest of the codes lead to ambiguous interpretations. For two bit errors the ambiguity is preserved while for three bit errors only 0000 and 1111 can reliably interpreted as "0" and "1". If we now look at the *Selected* coding scheme we can see that reliable interpretation can be made even when five errors are allowed (third row). The Gray codes behave very well when few errors are possible the *Selected* coding we developed behaves much better when there are more errors (noise) in our data. The previous comparison is meant as a reference to the patterns of both coding and not as a reference to their ability to transfer signals since in the *Selected* case we do not consider bit errors outside the displayed patterns. The *Selected* is just a very specialized subset of the Gray codes of equivalent length (eight in the case of table 2).

In the case of the previous table if one was to preserve the length of the Gray codes that is four we would end up with only the following elements for our *Selected* list:

0000, 0001, 0011, 0111, 1111, 1110, 1100, 1000

What is interesting in the *Selected* codes is that the number of errors are a linear function of their distance from each other. So in the previous arrangement if we take as reference the element 0011 we can see that its immediate neighbors 0001 and 0111 differ by one bit while the following neighbors (two positions apart) 0000 and 1111 differ by two bits. This continues until then maximum allowable number of error is reached. To better visualize the above property three *Selected* codes (00000111, 00000001, 11111111) are chosen from

Gray Code	Number of Errors				Number of Errors				
	1	2	3	**Selected Coding**	1	2	3	4	5
0000	0	0	0	**00000000**	0	0	0	0	0
0001	0	0	0 1	**00000001**	0	0	0	0	0
0011		0 1	0 1	**00000011**		0	0	0	0
0010	0	0	0 1	**00000111**			0	0	0 1
0110		0 1	0 1	**00001111**				0 1	0 1
0111	1	1	0 1	**00011111**			1	1	0 1
0101		0 1	0 1	**00111111**		1	1	1	1
0100	0	0	0 1	**01111111**	1	1	1	1	1
1000	0	0	0 1	**11111111**	1	1	1	1	1
1001		0 1	0 1	**11111110**	1	1	1	1	1
1011	1	1	0 , 1	**11111100**		1	1	1	1
1010		0 1	0 1	**11111000**		1	1	1	0 1
1110	1	1	0 1	**11110000**				0 1	0 1
1111	1	1	1	**11100000**			0	0	0 1
1101	1	1	0 1	**11000000**		0	0	0	0
1100		0 1	0 1	**10000000**	0	0	0	0	0

Tables 2. Comparative interpretation of Gray and *Selected* coding

table 2 and displayed in table 3 with the error bit count from the rest of the Selected codes. From the table it is evident the relationship between number of errors and distance from target code.

Selected Coding Errors	00000111	00000001	11111111
00000000	3	1	8
00000001	2	**0**	7
00000011	1	1	6
00000111	**0**	2	5
00001111	1	3	4
00011111	2	4	3
00111111	3	5	2
01111111	4	6	1
11111111	5	7	**0**
11111110	6	8	1
11111100	7	7	2
11111000	8	6	3
11110000	7	5	4
11100000	6	4	5
11000000	5	3	6
10000000	4	2	7

Table 3. Error distance between *Selected* codes.

The displayed property is extremely useful when one studies cyclic variables that are functions of the rotational angle and try to make predictions about its value. Tables 4 illustrated such a case where an accuracy of up to 5.625 degrees is sought. On can easily see that the difference of 0 and 354.38 degrees is only one bit. Similarly the difference of 180 and 135 degrees that is 8*5.625 is exactly eight bits exactly as much as their difference in their positions in the scheme.

Observing that ones and zeros appear in continuous segments it could be easily enhance the *Selected* coding with forward error correction. For example the case of:
11111111111111110000000000000000

Any zeros between the first and last "1" can be ignores and any ones between the first and last "0" can also be ignored. Problematic areas are only the transition areas between "1"s and "0"s. These will

Gray	Selected Coding	Angle
000000	01111111111111111111111111111111	0.00
000001	00111111111111111111111111111111	5.63
000011	00011111111111111111111111111111	11.25
000010	00001111111111111111111111111111	16.88
000110	00000111111111111111111111111111	22.50
000111	00000011111111111111111111111111	28.13
000101	00000001111111111111111111111111	33.75
000100	00000000111111111111111111111111	39.38
001100	00000000011111111111111111111111	45.00
001101	00000000001111111111111111111111	50.63
001111	00000000000111111111111111111111	56.25
001110	00000000000011111111111111111111	61.88
001010	00000000000001111111111111111111	67.50
001011	00000000000000111111111111111111	73.13
001001	00000000000000011111111111111111	78.75
001000	00000000000000001111111111111111	84.38
011000	00000000000000000111111111111111	90.00
011001	00000000000000000011111111111111	95.63
011011	00000000000000000001111111111111	101.25
011010	00000000000000000000111111111111	106.88
011110	00000000000000000000011111111111	112.50
011111	00000000000000000000001111111111	118.13
011101	00000000000000000000000111111111	123.75
011100	00000000000000000000000011111111	129.38
010100	00000000000000000000000001111111	135.00
010101	00000000000000000000000000111111	140.63
010111	00000000000000000000000000011111	146.25
010110	00000000000000000000000000001111	151.88
010010	00000000000000000000000000000111	157.50
010011	00000000000000000000000000000011	163.13
010001	00000000000000000000000000000001	168.75
010000	00000000000000000000000000000000	174.38
110000	10000000000000000000000000000000	180.00
110001	11000000000000000000000000000000	185.63
110011	11100000000000000000000000000000	191.25
110010	11110000000000000000000000000000	196.88
110110	11111000000000000000000000000000	202.50
110111	11111100000000000000000000000000	208.13
110101	11111110000000000000000000000000	213.75
110100	11111111000000000000000000000000	219.38
111100	11111111100000000000000000000000	225.00
111101	11111111110000000000000000000000	230.63
111111	11111111111000000000000000000000	236.25
111110	11111111111100000000000000000000	241.88
111010	11111111111110000000000000000000	247.50
111011	11111111111111000000000000000000	253.13
111001	11111111111111100000000000000000	258.75
111000	11111111111111110000000000000000	264.38
101000	11111111111111111000000000000000	270.00
101001	11111111111111111100000000000000	275.63
101011	11111111111111111110000000000000	281.25
101010	11111111111111111111000000000000	286.88
101110	11111111111111111111100000000000	292.50
101111	11111111111111111111110000000000	298.13
101101	11111111111111111111111000000000	303.75
101100	11111111111111111111111100000000	309.38
100100	11111111111111111111111110000000	315.00
100101	11111111111111111111111111000000	320.63
100111	11111111111111111111111111100000	326.25
100110	11111111111111111111111111110000	331.88
100010	11111111111111111111111111111000	337.50
100011	11111111111111111111111111111100	343.13
100001	11111111111111111111111111111110	348.75
100000	11111111111111111111111111111111	354.38

Table 4. Case study with their corresponding angles in degrees

be handled by very nature of the *Selected* coding that nearby values differ by few bits. Forward error correction will be useful in cases where there is a lot of noise and bit changes can appear in more than one position.

A Java implementation of the algorithm that produces the new coding scheme for code length *nBase* follows:

```
int nCodes = 2*nBase;
int SelectedCodes[][] = new int[nCodes][nBase];

for (int i=0; i<nCodes; i++)
   for (int j=0; j<nBase; j++)
      if (i < nCodes/2)
         if (j>=i) SelectedCodes[i][j] = 0;
         else SelectedCodes[i][j] = 1;
      else if (j+nBase>=i) SelectedCodes[i][j] = 1;
            else SelectedCodes[i][j] = 0;
```

3. Conclusions and Future Work

A coding scheme as a subset of Gray codes has been developed to express errors as a linear function of their respective code position. The scheme uses longer codes than typically used in Gray codes but tolerates more error per code. It can be easily implemented algorithmically and can be reliably used for pattern representation in neural networks and genetic algorithms. With forward error correction it can easily be used for signal transmission.

Future research will focus on applying the scheme in real life situations specifically in training neural networks and evaluate its performance to represent input patterns of rotation parameters. In addition existing forward error correction techniques will be applied and new ones will be investigated.

References:

[1] Goodall W.M., "Television by Pulse Code Modulation", *Bell Sys. Tech. J.*, Vol. 30, 1951, pp. 330-349.

[2] Richards D., "Data compression and Gray-code sorting," *Inform. Process. Lett.*, Vol. 22, 1986, pp. 201-205.

[3] Schwartz M., Etzion T., "The Structure of Single-Track Gray Codes", *IEEE Transactions*

on Information Theory", Vol. 45, No. 7, 1999, pp. 2383-2396.

[4] Bhat G.S., Savage C.D., "Balanced Gray codes," *Elec. J. Combin.*, Vol. 3, No. 1, p. R21, 1996.

[5] Savage C., "A survey of combinatorial Gray codes," *SIAM Rev.*, Vol. 39, 1997, pp. 605-629.

[6] Chang C C., Chen H.Y., Chen C.Y., "Symbolic Gray code as a data allocation scheme for two-disc systems," *Comput. J.*, Vol. 35, 1992, pp. 299-305.

[7] Diaconis P., Holmes S., "Gray codes for randomization procedures," *Statist. Comput.*, Vol. 4, 1994, pp. 287-302.

ISSN: 1790-2769

ISBN: 978-960-474-133-5