Revista de Ciências da Computação, Volume III, Ano III, 2008, nº3

Zero Displacement Ternary Number System: the most economical way of representing numbers

Fernando Guilherme Silvano Lobo Pimentel,
Bank of Portugal,
Email: fpimentel@bportugal.pt

Abstract

This paper concerns the efficiency of number systems. Following the identification of the most economical conventional integer number system, from a solid criteria, an improvement to such system's representation economy is proposed which combines the representation efficiency of positional number systems without 0 with the possibility of representing the number 0. A modification to base 3 without 0 makes it possible to obtain a new number system which, according to the identified optimization criteria, becomes the most economic among all integer ones.

Key Words: Positional Number Systems, Efficiency, Zero

Resumo

Este artigo aborda a questão da eficiência de sistemas de números. Partindo da identificação da mais económica base inteira de números de acordo com um critério preestabelecido, propõe-se um melhoramento à economia de representação nessa mesma base através da combinação da eficiência de representação de sistemas de números posicionais sem o zero com a possibilidade de representar o número zero. Uma modificação à base 3 sem zero permite a obtenção de um novo sistema de números que, de acordo com o critério de optimização identificado, é o sistema de representação mais económico entre os sistemas de números inteiros.

Palavras-Chave: Sistemas de Números Posicionais, Eficiência, Zero

1 Introduction

Counting systems are an indispensable tool in Computing Science. For reasons that are both technological and user friendliness, the performance of information processing depends heavily on the adopted numbering system. For example if we choose to perform arithmetics using only one symbol, a simple 100 + 100 addition would force us to write down that symbol two hundred times and look at the result to get an idea of how much that is. Such an approach might have been helpful in caveman times but unconceivable now for todays arithmetics.

Roman numerals are an example of how to use more symbols to overcome the difficulties associated with the one symbol approach. Representing small numbers with such notation is still used today in our clocks or book chapter numbers. However, when it comes to perform arithmetics, roman numerals aren't as good as ours. Though complicated, addition is still easy to accomplish. However, multiplication with Roman numerals is rather tedious. In fact, a good exercise left out to the reader would be to try to multiply roman numerals and find out how good our usual Hindu-Arabic numbers are.

The secret behind the arithmetics that we learn in school being so easy, lies in the emergence of the positional numbering systems. While roman numerals concatenate symbols in a way that each symbol is always worth the same number (X, for example, is always worth ten), positional numbering systems value each symbol contribution as a certain number according to its position in the digit sequence. Usually each position in the digit sequence represents the next power (from right to left and starting from 0) of a certain number, that number being called the radix of the base or numbering system. The symbol, occupying each position in the sequence, should therefore be multiplied by the corresponding power of the radix and all contributions, added together afterwards, to get the number.

For example in our common decimal base (radix = 10) we write the number 132 as the contribution of $2*10^0 + 3*10^1 + 1*10^2$ (reading the number from right to left)

To represent the same number in another base one can use the following algorithm [see for example Wikipedia contributors, November 2008] to obtain the sequence of numbers which, once reversed, gives the translation of the number in that base:

Divide the number by the radix, and the remainder is the least-significant digit. Then divide again the result by the radix, and collect the remainder as the next most significant digit, repeating the process until the result of further division becomes zero.

In base 6, for example, we would have:

132/6	= 22	, remainder = 0
22/6	= 3	, remainder = 4
3/6	= 0	, remainder = 3

Therefore in base 6 we would write 132 as 340.

To write all integers up to 132, both bases need only a maximum of 3 digits. However, base 10 uses ten symbols while base 6 needs only six. Therefore, one could say that to represent numbers up to 132, base 6 is more economic, or efficient, than base 10.

Generally, in positional numbering systems, a balance has to be reached between the number of different symbols (radix) and the number of digits (width) that are required to represent a number. That is, when one increases, the other decreases.

As extreme cases we would have the unary base – the one-symbol approach – with a minimum number of symbols but a maximum number of digits necessary to represent numbers and bases with an enormous number of symbols but very few digits necessary to represent a number like, for instance, the 1.000.000.000 base that would use only one digit for numbers up to 999.999.999. Obviously, either one would seem very unpractical, but the question of how we can determine whether or not we are using the most efficient numbering system or systems, still remains.

The specific positional number representation systems, or bases, that are mostly used today emerged from their operationally. Base 10, for example, is quite appropriate for counting with the fingers, which is commonly mentioned as the reason why it became so popular.

Meanwhile, the development of calculation mechanisms brought other technological considerations into place, and the binary (base 2) system became widely used in computers. In electronics, to distinguish between several states (for which it is necessary to model different symbols for each digit) it is much more difficult than to just distinguish between two of them.

However, no matter which physiological or technological circumstances we may face nowadays, one may still ask which is the most efficient numbering system available.

Given the fact that different numbering systems might hold different specific properties of their own (for example, in terms of their arithmetic) as is the case of the system that is introduced in this paper, if a given system is more efficient than others, then a motivation for our pursuit might be to wonder if nature exhibits in some of its mechanisms, characteristics that are similar to that system's properties, making it the numbering system that is appropriate to describe what is going on in some of its domain.

With this rather exploratory perspective in mind, we abstract, in this paper, from today's technological restrictions behind computing circuitry and proceed with the theoretical exercise of selecting a mathematical measure of the efficiency of a numbering system, identify the most convenient known system from that measure's point of view, and search for any possible improvements in such system's efficiency.

2 Efficiency of number representation systems

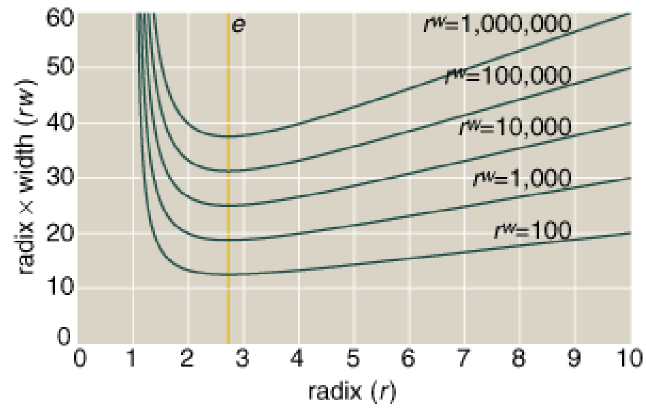
At least two different strategies have been proposed to define some cost measure for numbering systems and determine which ones are the most efficient.

The first focused on minimizing the sum of the total number of different entities whose names the human mind must remember [Phythian 1969]. They are the number of digits together with the number of place values. For instance, in base seven one would need names for the digits 0 1 2 3 4 5 6 and the powers 7, 7^2, 7^3, …

As the numbers to be represented increase, so does the number of names required to represent them and the more adequate number base for that purpose. For example, to represent numbers up to 15, in base two, one needs only 5 names, while, in base seven, 8 names would be necessary. However, to represent numbers up to 10000, in base two, one needs 15 names while, in base seven, 11 names would be enough. It turns up that the most economical base to represent numbers up to a certain value depends on that range of values, increasing with it in a way that can be calculated [Phythian 1969].

The variation of efficiency of each base accordingly to the numbers to be represented, is a point that goes against the strategy of using the sum of the number of digits together with the number of place values as the cost measure of numbering systems. In fact, since changing between numbering systems according to the magnitude of the numbers to be represented is not viable, one looks for a stable solution, no matter how big are the numbers to be represented.

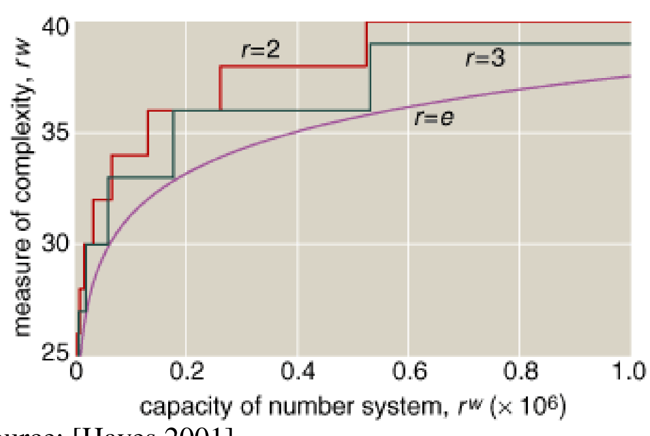
This opens up room for the second strategy of minimizing, not the sum, but the product between the number of different symbols used and the number of digits required to represent a number. This way, the efficiency measure of each system would be directly proportional to any of these two quantities. Such strategy to obtain the most economical base has been reported in the literature [Hayes 2001]. Treating the two variables as continuous, it turns out that, no matter how big the number to be represented, the most efficient base is e [Hayes 2001].



Source: [Hayes 2001]

Figure 1. Most economical radix for a numbering system is e (about 2.718) when economy is measured as the product of the radix (base or number of symbols) and the width, or number of digits, needed to express a given range of values. Here both the radix and the width are treated as continuous variables.

Building on the integer value that is closest to e, base 3 (using 0,1 and 2) is almost always, according to this strategy, the most economical integer base, as represented in Figure 2.



Source: [Hayes 2001]

Figure 2. The most economical integer radix is almost always 3, the integer closest to e. If the capacity of a numbering system is r^w, and the cost of a representation is rw, then r=3 is the best integer radix for all but a finite set of capacities. Specifically, ternary is inferior to binary only for 8,487 values of r^w; ternary is superior for infinitely many values.

3 Number representation systems without zero

In spite of the fact that normal ternary base has apparently emerged as the most economical integer base, according to the selected efficiency criteria, there is a better performing number representation system. Even with the shortcoming of not representing zero, a modification of base 3 from using 0,1 and 2 to using 1, 2, and 3 instead, would lead to an increase of efficiency. Table 1 shows the representation of the first 40 non-negative integers in those bases. It can be noted that one digit is saved in almost half of the numbers.

Table 1 – The first 40 non-negative integers in base 10, base 3 (0,1,2) and base 3 (1,2,3)

Base 10	Base 3 (0,1,2)	Base 3 (1,2,3)
0	0	
1	1	1
2	2	2
3	10	3
4	11	11
5	12	12
6	20	13
7	21	21
8	22	22
9	100	23
10	101	31
11	102	32
12	110	33
13	111	111
14	112	112
15	120	113
16	121	121
17	122	122
18	200	123
19	201	131

Base 10	Base 3 (0,1,2)	Base 3 (1,2,3)
20	202	132
21	210	133
22	211	211
23	212	212
24	220	213
25	221	221
26	222	222
27	1000	223
28	1001	231
29	1002	232
30	1010	233
31	1011	311
32	1012	312
33	1020	313
34	1021	321
35	1022	322
36	1100	323
37	1101	331
38	1102	332
39	1110	333

Number representation systems without zero have already been reported in the literature (see for example [Foster 1947], and [Forslund 1995]). The advantage of such systems needing, oftenly, less digits to represent a number has been mentioned [Forslund 1995]. However, one important issue remained to be solved: the representation of 0. In fact, as its historical uses have shown, the number zero is important not only as an empty place indicator in our place-value numbering system, but also as a number itself [O'Connor 2000]. In fact, for a number system to be considered as perfect, one of the criteria is that no number should go unrepresented [Bhattacharjee, 1995].

Number representation systems without zero have also been called Bijective Numeration Systems or k-adic notation systems. Bijective base-k numeration represents a non-negative integer by using a string of digits from the set $\{1, 2, ..., k\}(k \geq 1)$ to encode the integer's expansion in powers of k [Wikipedia contributors, September 2008]. The fact that every non-negative integer has a unique representation in such system - bijective base-k ($k \geq 1$) is known [Wikipedia contributors, 2008] and several particular cases (in terms of k's domain) have been previously mentioned in the literature ([Böhm 1964], [Knuth 1969], [Salomaa 1973] and [Smullyan 1961] - cited in [Wikipedia contributors, 2008]).

Bijective Numeration Systems represent the integer zero by using the empty string, but one needs then one extra symbol to allow for such representation, which clearly reduces its efficiency.

Therefore, the question remains of how to obtain the representation economy gains of a positional number system without zero, while maintaining the possibility of representing 0 in such system.

4 A new number representation system

To present the new number representation system one considers the case of base 3, since it is the purpose of this paper to identify the most economical number representation system available, according to the criteria identified in the literature review. The idea, however, is obviously extensible to other bases.

For the purpose of illustrating the representation of the first 40 non-negative integers in the Zero Displacement Ternary Number System, table 2 shows how it compares with the other representations. In table 2, among the bases that allow 0 to be represented, Zero Displacement Ternary Number System (ZDTNS) is the most economic one.

Table 2 – The first 40 non-negative integers in base 10, base 3 (0,1,2), base 3 (1,2,3) and ZDTNS

Base 10	**Base 3 (0,1,2)**	**Base 3 (1,2,3)**	**ZDTNS**	**Base 10**	**Base 3 (0,1,2)**	**Base 3 (1,2,3)**	**ZDTNS**
0	0		1	20	202	132	133
1	1	1	2	21	210	133	211
2	2	2	3	22	211	211	212
3	10	3	11	23	212	212	213
4	11	11	12	24	220	213	221
5	12	12	13	25	221	221	222
6	20	13	21	26	222	222	223
7	21	21	22	27	1000	223	231
8	22	22	23	28	1001	231	232
9	100	23	31	29	1002	232	233
10	101	31	32	30	1010	233	311
11	102	32	33	31	1011	311	312
12	110	33	111	32	1012	312	313
13	111	111	112	33	1020	313	321
14	112	112	113	34	1021	321	322
15	120	113	121	35	1022	322	323
16	121	121	122	36	1100	323	331
17	122	122	123	37	1101	331	332
18	200	123	131	38	1102	332	333
19	201	131	132	39	1110	333	1111

Among all the numbers that are smaller than n, there are k-1 numbers (with k = number of digits necessary to represent n in base 3 with 1,2,3) for which the representation in ZDTNS needs one more digit than in Base 3 (1,2,3) – for example in the first 40 non-negative integers there are 3 such numbers (3, 33 and 333).

Except for this, and while allowing 0 to be represented, the displacement number system basically holds the same efficiency as base 3 with 1,2,3 (Forslund system). In fact, the number of numbers with n digits that can be written in the Zero Displacement Ternary Number System is the same as in base 3 (1,2,3), as table 3 illustrates.

Table 3 – Representation of how many numbers with n digits can be written in different number systems for the first 40 non-negative integers

n (number of digits)	Number of numbers that can be written	
	Base 3 (1,2,3) or Forslund System	Zero Displacement Ternary Number System
1	3	3
2	9	9
3	27	27

The translation mechanism from and to Zero Displacement Ternary Number System is quite simple. Consider for example that: a) we want to write a decimal base number n in ZDTNS; and b) we want to translate a ZDTNS number j to decimal base:

a) To represent an integer number n (positive, negative or zero) in the Zero Displacement Ternary Number System (ZDTNS) one proceeds as follows:

1. If n is different from 0 Then Add $\frac{|n|}{n}$ to the number n
   Else Add 1 to the number n

2. Represent the value previously computed in base 3 without zero (1,2,3).

b) To read number j from Zero Displacement Ternary Number System one proceeds as follows:

1. Take number j in base 3 without zero (1, 2, 3), subtract $\frac{|j|}{j}$ from it and then translate the result from that base to base 10.

In Zero Displacement Ternary Number System (ZDTNS), the arithmetic is different from what is usual in conventional numbering systems. As a separate issue in itself, its treatment is to be focused in another work

In future developments, besides its basic arithmetic, several other aspects of this numbering system are to be detailed in future work such as the system formalization in axiomatic terms, analysis of its properties, generalization to fractional numbers, possible applications, etc.

5 Conclusion

Following the identification of an appropriate mathematical measure of the efficiency of a number system - that consists of the product between the number of different symbols and the number of digits used to represent a number – we have verified how base three is reported as the most economic conventional number system from that measure's point of view.

The search for possible improvements in base three efficiency led us to the consideration of number systems without zero, which in fact allow gains in economy but have the disadvantage of not representing number zero.

As an original contribution to number systems in general and to the goal of identifying the most efficient number representation system in particular, a solution is proposed to the problem of zero representation in number systems without zero. As a result, the Zero Displacement concept is proposed and its translation mechanism between bases introduced for the Zero Displacement Ternary Number System, which was identified as the most economical number representation system, strictly from the point of view of number representation.

References

Bhattacharjee, Abhijit, 1995. A polar place value number system. Available in: http://abhijit.info/ tristate/tristate.html#numsystem

Böhm, C., 1964. On a family of Turing machines and the related programming language", ICC Bulletin 3, p. 191.

Forslund, Robert, 1995. A Logical Alternative to the Existing Positional Number System. Southwest Journal of Pure and Applied Mathematics, Volume 01, pp. 27-29.
Available in: http://my.tbaytel.net/forslund/rrf01.html

Foster, James, 1947. A Number System without a Zero-Symbol. Mathematics Magazine, Vol. 21, No.1, pp. 39-41.

Hayes, Brian, 2001. Third Base. A reprint from American Scientist, Volume 89, Number 6, pp. 490-494.

Knuth, D. E., 1969. The Art of Computer Programming, Vol. 2: Seminumerical Algorithms, 1st ed., Addison-Wesley, p. 495.

O'Connor, J.J., Robertson, E.F., 2000. A history of Zero in "Ancient Indian Mathematics". JOC/ EFR. Available in: http://www-history.mcs. st-andrews.ac.uk/ HistTopics/Zero.html

Phythian, J. E., 1969. Economical Number Bases. The Mathematical Gazette, Volume 53, Number 386, pp. 376-379.

Salomaa, A., 1973. Formal Languages, Academic Press., pp. 90-91.

Smullyan, R., 1961. Theory of Formal Systems, Annals of Mathematics Studies, Number 47, Princeton.

Wikipedia contributors, (2008, September 10). Bijective numeration. Wikipedia, The Free Encyclopedia. Retrieved 10:23, October 14, 2008, from http://en.wikipedia.org/w/index.php?title=Bijective_numeration&oldid=237542776

Wikipedia contributors, (2008, November 8). Binary numeral system. Wikipedia, The Free Encyclopedia. Retrieved 15:08, November 13, 2008, from http://en.wikipedia.org/w/index.php?title=Binary_numeral_system&oldid=250449643