Primary Study of the Calculating Speed of Natural Number in Different Scale Systems

Team member: Ye Tianyang Wei Bo Wang Mingyuan

Teacher: Ling Huiming

School: Jinling High School, Nan Jing

Abstract

This article studied the speed of addition and multiplication of natural number in different scale systems, the times of addition and multiplication in different scale systems have been compared quantificationally through a function model presented. The conclusion is that binary system is the best for addition, binary system and ternary system are better than other scale systems for multiplication, and they each has a suitable range.

Introduction

In modern society, large quantities of data are dealt with everyday, these all need computers. In the development of digital computers, binary system played a very important role. No matter how complex the calculation in a computer is, it is all completed by the binary addition of digital circuit. The two steady states in digital circuit represent ZERO and ONE in binary system. With the development of technology, man may create computers supported by electronic devices which have three or more than three steady states, at that time computers do not have to be based on binary system. If the scale system was chosen properly, the time of calculation could be reduced. Compared with all the scale systems, it is obvious that a big module scale system is more convenient to express a number, and does less operations, unfortunately the operations are complicated, and more symbols are needed; a small module scale system is easy to do operations, but difficult to express the numbers and does more operations. This paper aimed at presenting a function to compare the calculating speed of natural numbers in different scale systems, finding the most suitable scale system for a certain range of numbers.

Considering the complexity of this problem, only the addition and multiplication of natural numbers are discussed here.

In this paper k represents natural numbers larger than 2.

Since this paper is discussing the character of scale systems in wide range of calculating, probabilities are used.

NUMBER is an abstract existence in natural, man expressed it with a symbol ,that's the numbers people deal with every day, operations in a certain scale system is to transform the symbols according to certain rules (according to the

addition or multiplication table in this scale system), the number which the transformed symbol represents is the result of this operation.

Then the most basic operation is considered, the addition or multiplication of two one digit numbers, it is called unit operation.

As said above, the process of operating is the transforming of symbols, it can be said that the process of unit operation is a process of looking up the table. Operator look for the line and row of the addend (multiplicator) in the addition (multiplication) table, the number at the intersection is the result of this operation. To treat everybody impartially, it is assumed that the speed of looking up table is the same to all the scale systems, and then time of operation is in direct proportion to the size of addition (multiplication) table. Furthermore, since addition and multiplication were taken as the transforming of symbols by looking up table, what the contents of the table is does not affect the time, so the addition and multiplication of one digit numbers take the same time in a certain scale system. It's easy to know that the size of addition (multiplication) table in module *k* scale system is $k \times k$ (including zero). In this paper the time of looking up the table in "module 1 scale system" is thought as unit time t, and then the time of unit operation in module k scale system is

$$t_k = k^2 t \qquad (1)$$

It is also know, if A expressed as an m digit number in module k system, then

$$k^{m-1} \le A \le k^m - 1 \quad m = \left[\log_k A\right] + 1 \qquad (2)$$

1. Addition

In module *k* scale system, an *m* digit number plus an *n* digit number, let $n \le m$

To study the amount of operations, the normal formula is used.

But here another order is used to do the operations. There is no carry at first, just plus the numbers above and below, and then add the carries to the result.

For example: calculate 678+125.

$$\begin{array}{rrr} & 6 & 7 & 8 \\ + & 1 & 2 & 5 \\ \hline \end{array}$$

The usual order of formula is 8+5=13 carry 1

2+7=9 9+1=10 carry 1

6+1=7 7+1=8

Here the order is 8+5=13 7+2=9 6+1=7

carry 1 9+1=10 first passel of carries

carry1 1+7=8 second passel of carries

Since it could only carry 1 in addition, when two numbers add together and then add the carry 1, there is no possibility to carry again $\left((k-1)+(k-1)+1<2k\right)$, the amount of operation stays the same in different orders, the order of operation does not affect the time of operation.

An *m* digit number add an *n* digit number, first you have to do $l_1 = n$ times of unit operations, and then the amount of carries is considered.

Considering the carries get from the n times of unit operations, the first passel of carries(see explanations besides the formulas), two one digit numbers, *a* plus *b*, because of symmetry , the probability for *a* or *b* to be one of 0,1,…(*k*-1) is the same. If *a* plus *b* does not carry, then a+b must be one of 0, 1… (*k*-1). The amount of the nonnegative integer solutions of the indefinite equation $a+b=p \quad 0 \le p \le k-1$ is $\sum_{p=0}^{k-1} C_{p+2-1}^{2-1} = \sum_{p=0}^{k-1} C_{p+1}^{1} = C_{\mathrm{k}+1}^{2}$.

Considering the order pair of (a,b), it consists k^2 elements.

So the probabilities that do not carry is $p(do\ not\ carry) = \frac{C_{k+1}^2}{k^2} = \frac{k+1}{2k}$

The probabilities that carries is $p(the\ first\ carry) = 1 - \frac{k+1}{2k} = \frac{k-1}{2k}$

So there are $n(\frac{k-1}{2k})$ carries in the first n unit operations.

Then consider the second passel of carries. The second passel of carries are different from the first ones for it can only get 1 from the first carries, they must be in the form of c+1. Because of symmetry, the probability for c to be one of 0,1,…(k-1) is the same

(Take decimal system for example. It can get 1 as the last number by plus

（0，1）（1，0）（2，9）（9，2）（3，8）（8，3）（4，7）（7，4）（5，6）（6，5）

And it can get 2 by plus

（0，2）（2，0）（1，1）（3，9）（9，3）（4，8）（8，4）（5，7）（7，5）（6，6）

The amounts of situations are the same)

Therefore, there is a carry if and only if c=k-1

So the probability for the second passel of carries to carry is

$$p(\mathrm{sec}\,ond\ \ carry) = \frac{1}{\mathrm{k}}$$

Therefore, there are $n(\frac{k-1}{2k})\frac{1}{\mathrm{k}}$ carries in the second passel of carries.

In the same way，the probability for third, fourth passel of carries to carry is also $\frac{1}{\mathrm{k}}$.

The sum of carries is $\sum_{i=0}^{\infty} n(\frac{k-1}{2k})(\frac{1}{k})^i = n(\frac{k-1}{2k})(\frac{1}{1-\frac{1}{k}}) = n(\frac{k-1}{2k})(\frac{k}{k-1}) = \frac{n}{2}$

So the carries resulted in $l_2 = \frac{n}{2}$ times of operations.

Therefore an *m* digit number plus an *n* digit number need operations of

$$l_1 + l_2 = n + \frac{n}{2} = \frac{3n}{2} \qquad (3)$$

times.

This is an interesting conclusion, for it only depends on the digit number not *k*.

For two numbers A, B, $A \le B$, from (1)(2)(3), it needs $\frac{3([\log_k A]+1)}{2}$ times of operations in module k scale system, and it needs time

$$\frac{3([\log_k A]+1)}{2} k^2 t \qquad (4)$$

For a certain range of A, look for the *k* that needs the least time. This function is pretty complex for *k* but much easier for A. Here think from the other side, for a certain *k*, look for A that suits it best.

If for a certain range of A, module *k* scale system takes less time then module *k*+1 scale system to do operation, then

$$\frac{3([\log_k A]+1)}{2} k^2 t \le \frac{3([\log_{k+1} A]+1)}{2} (k+1)^2 t$$
$$\Leftrightarrow ([\log_k A]+1)k^2 \le ([\log_{k+1} A]+1)(k+1)^2$$

Here a approximation was made，replace the discrete function with a continuous function

Let $[\log_k A] \approx \log_k A = \frac{\ln A}{\ln k}$ $[\log_{k+1} A] \approx \log_{k+1} A = \frac{\ln A}{\ln(k+1)}$

Then $([\log_k A]+1)k^2 \le ([\log_{k+1} A]+1)(k+1)^2$

$$\Leftrightarrow \ln A\left(\frac{k^2}{\ln k} - \frac{(k+1)^2}{\ln(k+1)}\right) \le 2k+1$$

$$let \quad f(x) = \frac{x^2}{\ln x} \quad then \quad f'(x) = \frac{2x\ln x - \frac{1}{x}\times x^2}{(\ln x)^2} = \frac{x(2\ln x - 1)}{(\ln x)^2} > 0 \quad \forall x \in [2,+\infty)$$

So for $x \in [2,+\infty)$ $f(x)$ is an increasing function.

Therefore, $\left(\frac{k^2}{\ln k} - \frac{(k+1)^2}{\ln(k+1)}\right) < 0$

So the left hand of the equivalent inequality is a negative number and the right hand of it is a positive number, the inequality is obviously true.

Let t(*k*) is the time of addition in module *k* scale system, and then from the

analyses above it could be known $for\ \forall A \in N^*\quad t(2) \le t(3) \le \cdots \le t(k)$

So the conclusion is that binary system is the best for addition.

2. Multiplication

Starting from the simplest situation, consider a one digit number multiple a *m* digit number。

Firstly, it is needed to do m times of unit operation. The result might be one digit number or two digit number. Let the probability that the result is two digit number in module *k* system be $p_{k,}$ it could be figured out through the multiplication table. Table 1 shows some p_k we worked out with the help of computer.

Table 1 *k* and p_k

k	p_k	k	p_k	k	p_k	k	p_k
2	0	7	0.448979592	12	0.638888889	17	0.712802768
3	0.111111111	8	0.515625	13	0.644970414	18	0.731481482
4	0.25	9	0.543209877	14	0.673469388	19	0.736842105
5	0.32	10	0.58	15	0.688888889	20	0.7525
6	0.416666667	11	0.603305785	16	0.703125	21	0.757369615

When *k* is quite a large number, consider the situation that the result is one digit number through the multiplication table showed in table 2, and find the approximation

Table 2 the multiplication table in module *k* scale system

×	0	1	2	…	k-2	k-1
0	0	0	0		0	0
1	0	1	2		k-2	k-1
2	0	2	4		2k-4	2k-2
…						
k-2	0	k-2	2k-4			
k-1	0	k-1	2k-2			

of p_k. It is easy to know that the numbers in the zero line($0 \times A$) all do not carry, the number in the one line ($1 \times A$) all do not carry, and there are $\left[\frac{k-1}{2}\right]+1$ numbers in the two line ($2 \times A$) that do not carry, like the approximation did before, replace the discrete function with continuous function , let $\left[\frac{k-1}{2}\right]+1 \approx \frac{k}{2}$, and so forth。

Then the probability that do not carry is

$$\bar{p}_k = 1 - p_k = \frac{1}{k^2}\left(k + k + \left[\frac{k-1}{2}\right] + 1 + \left[\frac{k-1}{3}\right] + 1 + \cdots + \left[\frac{k-1}{k-2}\right] + 1 + \left[\frac{k-1}{k-1}\right] + 1\right)$$

$$\approx \frac{1}{k^2}\left(k + k + \frac{k}{2} + \frac{k}{3} + \cdots + \frac{k}{k-2} + \frac{k}{k-1}\right)$$

$$= \frac{1}{k}\left(2 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{k-2} + \frac{1}{k-1}\right)$$

It is also known $\int_1^{k-1} \frac{1}{x}dx > \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{k-2} + \frac{1}{k-1} > \int_1^{k} \frac{1}{x}dx - 1$

$$\Leftrightarrow \ln(k-1) > \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{k-2} + \frac{1}{k-1} > \ln k - 1$$

From this the interval of p_k can be figured out,

When k is quite large, let $p_k = 1 - \frac{\ln k + \ln(k-1) + 3}{2k}$, which is also the middle number of the interval.

					m digit number
*				9	1 *digit number*
			8	1	
+		7	2		
		8	0	1	
+	6	3			
	7	1	0	1	
+	5	4			
	6	1	1	0	1

					m digit number
*				1	1 *digit number*
				9	
+			8		
			8	9	
+		7			
		7	8	9	
+	6				
	6	7	8	9	

Figure 1 the process of adding up the results of one digit number multiply *m* digit number

Then consider the process of adding up the results. Here the usual order of addition is abandoned, instead add the outcomes one by one（showed in figure 1）, this does not affect the amount of operations. It is seen that the last number comes down directly, for one digit number and two digit numbers, the digit valuable is zero or one. It is needed to do m times of addition, and there are m-1 addends，from the analyses above it is known that there are $p_k(m-1)$ two digit numbers ，$(1-p_k)(m-1)$ one digit number. From (3) known, it need operations

$$\frac{3}{2} \times 1 \times p_k(m-1) + \frac{3}{2} \times 0 \times (1 - p_k)(m-1) = \frac{3}{2} p_k(m-1) \quad \text{times.}$$

In one word, one digit number multiple *m* digit number, needed to do multiplications m times, additions $\frac{3}{2} p_k(m-1)$ times

Then it is analyzed that an *m* digit number multiple an *n* digit number in module *k* scale system.

Theoretically，the final function must be a cyclic multinomial of m ,n.

Firstly, there are *mn* times of multiplications, this equals to do n times the operation that a one digit number multiple an *m* digit number, it needs multiplications $l_1 = mn$ times, additions $l_2 = n \times \frac{3}{2} p_k (m-1)$ times.

	1	1	1	1	1	（m digit ）
*			1	1	1	（n digit ）
	1	1	1	1	1	*n*
1	1	1	1	1		*numbers*
1	1	1	1	1		

Figure 2 the additions

Then looking through the additions below, there are n numbers, n-1 addends, (see figure 2) they need additions n-1 times. One digit number multiply *m* digit number may get *m* digit number or *m*+1 digit number, when the last number come straight down, the valuable digit is *m*-1 or *m*. Here a simplification is made, it is assumed that the probability that one digit number multiply *m* digit number will carry is also p_k (actually ,this probability should be larger than p_k, since the first figure of *m* digit number can not be zero, but this approximation is still allowable，especially when it is found that the final function is a cyclic multinomial of m ,n, accord with the theoretical need).

So there are p_k（n-1） *m*+1 digit number （1-p_k）（n-1）*m* digit number. From (3) known, in the additions it needs operations

$$l_3 = \frac{3m}{2} p_k (n-1) + \frac{3(m-1)}{2} (1-p_k)(n-1)$$
$$= \frac{3(n-1)}{2} \left[p_k m + (1-p_k)(m-1) \right] = \frac{3}{2} (n-1)(m-1+p_k)$$

times.

In one word，*m* digit number multiply *n* digit number needs operations

$$l_1 + l_2 + l_3 = mn + \frac{3n}{2} p_k (m-1) + \frac{3}{2} (n-1)(m-1+p_k)$$
$$= mn(1+\frac{3}{2} p_k) + \frac{3}{2} (m-1)(n-1) - \frac{3}{2} p_k \qquad (5)$$

It is a cyclic multinomial of m, n.

So, from (2)(5) known ,for two number A,B to multiply in module *k* scale system, it takes operation

$$([\log_k A]+1)([\log_k B]+1)(1+\frac{3}{2} p_k) + \frac{3}{2} ([\log_k A])([\log_k B]) - \frac{3}{2} p_k$$

times.

From (1) known, the operations take time

$$\left[([\log_k A]+1)([\log_k B]+1)(1+\frac{3}{2} p_k) + \frac{3}{2} ([\log_k A])([\log_k B]) - \frac{3}{2} p_k \right] k^2 t \qquad (6)$$

And in real situations, A, B usually come from the same certain range, to simplify the problem, let A=B.

Then replace the discrete function with a continuous function

$$[\log_k A] \approx \log_k A = \frac{\ln A}{\ln k}$$

The time for A multiple B is

$$\left[(\frac{\ln A}{\ln k}+1)^2 (1+\frac{3}{2}p_k) + \frac{3}{2}(\frac{\ln A}{\ln k})^2 - \frac{3}{2}p_k \right] k^2 t \qquad (7)$$

Just as did above, for a certain k look for A that suits it best.

From (7) known, if k suits better than k+1 for A, then for this range of A

$$\left[(\frac{\ln A}{\ln k}+1)^2 (1+\frac{3}{2}p_k) + \frac{3}{2}(\frac{\ln A}{\ln k})^2 - \frac{3}{2}p_k \right] k^2 t \le$$
$$\left[(\frac{\ln A}{\ln(k+1)}+1)^2 (1+\frac{3}{2}p_{k+1}) + \frac{3}{2}(\frac{\ln A}{\ln(k+1)})^2 - \frac{3}{2}p_{k+1} \right] (k+1)^2 t$$

$$\Leftrightarrow \left[(\frac{\ln A}{\ln k})^2 (\frac{5}{2}+\frac{3}{2}p_k) + 2(\frac{\ln A}{\ln k})\ (1+\frac{3}{2}p_k) + 1 \right] k^2 \le$$
$$\left[(\frac{\ln A}{\ln(k+1)})^2 (\frac{5}{2}+\frac{3}{2}p_{k+1}) + 2(\frac{\ln A}{\ln(k+1)})\ (1+\frac{3}{2}p_{k+1}) + 1 \right] (k+1)^2$$

$$\Leftrightarrow \left[\frac{(\frac{5}{2}+\frac{3}{2}p_k)k^2}{(\ln k)^2} - \frac{(\frac{5}{2}+\frac{3}{2}p_{k+1})(k+1)^2}{(\ln k+1)^2} \right] (\ln A)^2 + 2\left[(\frac{1+\frac{3}{2}p_k}{\ln k})k^2 - (\frac{1+\frac{3}{2}p_{k+1}}{\ln k+1})(k+1)^2 \right] \ln A \le 2k+1$$

Take the comparison between binary system and ternary system for example.

Let k=2, since k is not large, p_k use the exact value.

$$p_2 = 0, p_3 = \frac{1}{9}$$
$$\left[\frac{(\frac{5}{2}+\frac{3}{2}p_2)2^2}{(\ln 2)^2} - \frac{(\frac{5}{2}+\frac{3}{2}p_3)3^2}{(\ln 3)^2} \right] (\ln A)^2 + 2\left[(\frac{1+\frac{3}{2}p_2}{\ln 2})2^2 - (\frac{1+\frac{3}{2}p_3}{\ln 3})3^2 \right] \ln A \le 2\times 2+1$$

$$\Leftrightarrow (20.8136-19.8848)(\ln A)^2 + 2(5.7707-9.5575)\ln A \le 5$$
$$\Leftrightarrow -0.70113 \le \ln A \le 8.8548$$
$$\Leftrightarrow 0.49602 \le A \le 7008.112$$

（because we did not take much effective figures, the result is a little different from what the computer got, in this paper we admit the number from the computer）

From above, the range that binary system suits better than ternary system is figured out

$$0.5411988944 \le A \le 6423.1009727330 \qquad (8)$$

Afterward, the same method is done to find the suitable range for other scale systems (see table 3).

Table 3 the comparison between different k（p_k use the exact figure）

2 compare with 3	0.5411988944 <=A<= 6423.1009727330
3 compare with 4	0.0911847947 <=A<= 0.4859988020
4 compare with 5	0.1308296590 <=A<= 0.3997087792
5 compare with 6	0.0981193423 <=A<= 0.4703136977
6 compare with 7	0.1171215877 <=A<= 0.3655654868
7 compare with 8	0.0754583260 <=A<= 0.4555899733
8 compare with 9	0.0834543192 <=A<= 0.3847067303
9 compare with 10	0.0656224679 <=A<= 0.4157047754
10 compare with 11	0.0646738915 <=A<= 0.3876011628
11 compare with 12	0.0493320983 <=A<= 0.4276332168
12 compare with 13	0.0633592585 <=A<= 0.3409315275

From table 3 it can be seen, besides the comparison between binary system and ternary system, other comparisons do not have practical value. Here another way is tried to compare the scale systems, some representable A are chosen, for different k , calculate the operating time according to formula (7). The result did by computer is showed below in a table (table 4). Since unit time does not affect the comparison, here let t=1.

Table 4 the time table of different k and A（p_k use the exact figure）

A \ k	2	3	4	5	6	7	8	9	10
10	141	158	216	283	372	461	572	684	811
100	499	519	670	847	1080	1308	1594	1877	2196
1000	1077	1090	1377	1715	2159	2591	3130	3659	4255
10000	1876	1872	2339	2888	3610	4309	5179	6031	6988
100000	2896	2865	3554	4367	5433	6463	7743	8993	10395
1000000	4136	4068	5023	6150	7627	9052	10820	12545	14476
10000000	5597	5483	6746	8238	10192	12076	14410	16686	19231
100000000	7279	7108	8723	10631	13129	15536	18515	21417	24660
1000000000	9182	8945	10953	13329	16438	19432	23133	26738	30763
10000000000	11305	10992	13437	16333	20119	23762	28265	32649	37540

From table 4 it can be seen, in the interval analyzed, binary system and ternary system takes much less time than other scale systems , and from (8) known , binary system and ternary system each has a suitable range.

Considering the time in binary system is nearly the same as in ternary system, we

think binary system and ternary system both are the best for multiplications.

Conclusions

This paper discussed the speed of addition and multiplication in different scale systems, and compared the time it takes in different scale systems quantificationally. The conclusion is that, binary system is the best for addition, binary system and ternary system are all better than other scale systems in multiplications, and ternary system is better than binary system when multiplicator is larger than 6424. Considering that binary system and ternary system are the scale systems with the simplest symbol system, they are really worthy of the name of best scale systems. It is demonstrated that computers use binary system as the basic of calculation not only considered the steadiness of digital circuit, but also have advantages in scale systems.