DOI 10.1007/s10559-014-9654-0
Cybernetics and Systems Analysis, Vol. 50, No. 5, September, 2014

CYBERNETICS

THEORETICAL FOUNDATIONS FOR THE ANALYTICAL COMPUTATION OF COEFFICIENTS OF BASIC NUMBERS OF KRESTENSON'S TRANSFORMATION

Ya. M. Nykolaychuk,[a†] **M. M. Kasianchuk,**[a‡] **and I. Z. Yakymenko**[a††] UDC 519.7

Abstract. *This paper presents theoretical foundations for the analytical transformation of coefficients of basic numbers of Krestenson's transformation, which significantly reduces the number of operations required to convert numbers from a residue number system to the decimal number system. An appropriate selection of modules makes it possible to efficiently use all processor registers.*

Keywords: *residue number system, system of modules, basic number, Krestenson transformation, number-theoretical basis.*

INTRODUCTION

At present, one of main trends in developing computer aids is the creation of high-performance computing devices [1]. It is stipulated by the need for the solution of problems that are very important for the theory and practice of mathematics and require computations with integer multidigit numbers or quantities changing in rather large ranges [2].

In this connection, the applied and computational aspects of number theory are intensively being developing and are used in egineering systems to provide the reliability of transmission, storage, and processing of digital information. This leads to the need for the solution of many problems when computations arise in which lengths of integer variables can considerably exceed the format of existing universal computational tools, which is especially urgent with developing cryptographic methods and information protection facilities [3, 4].

ANALYSIS OF PUBLICATIONS

Any computational structure is closely connected with number-theoretic bases (NTBs) in which methods are given for coding (representing) elements of some finite model of real numbers by elements of one or several alphabets [5].

Arithmetic properties of any number system that are generated by the corresponding NTB are first of all determined by the nature of the interbit relations arising during the execution of the corresponding arithmetical and logical operations [6]. Investigations show [5, 7] that, within the scope of usual (decimal and binary) positional notations, it is practically impossible to reach a stepwise speedup of execution of the arithmetic operations of addition, subtraction, and multiplication. This results from the fact that the value of each digit of any number except for the least significant digit depends on values of not only the operands with the same name but also on all less significant digits, i.e., a positional notation possesses a rigorously sequential structure and requires the execution of through carries whose number is proportional to the register length in a processor. Thus, the use of positional notations leads to a considerable decrease in the speed and an increase in the computational and time complexities of algorithms being used. This especially implies the urgency of increasing the efficiency of processing multidigit numbers [2].

[a]Ternopil National Economic University, Ternopil, Ukraine, [†]*kmm@tneu.edu.ua*; [‡]*kasyanchuk@ukr.net*; [††]*jiz@tneu.edu.ua*. Translated from Kibernetika i Sistemnyi Analiz, No. 5, pp. 3–8, September–October, 2014. Original article submitted December 03, 2013.

Thus, the computational power of modern computers may be insufficient to solve many scientific, technical, and applied problems. Despite the fact that resources of newest computer facilities that function in positional notations are constantly being improved and increased, they basically cannot be boundless. This means that, within the scope of the performance provided by modern computer systems, broad classes of existing methods and algorithms cannot be implemented in practice, i.e., at present, positional notations exhaust their capabilities for constructing high-performance computer systems. A fundamental strategy of theoretical and practical investigations consists of the use of approaches based on the wide application of different forms of parallelism in computer systems. Nonpositional codes with parallel structures possess this distinctive feature. Among them, residue number systems (RNSs) that make it possible to implement the idea of parallelization at the level of execution of the elementary arithmetic operations of addition, subtraction, and multiplication are most promising [7, 8]. The representation of operands in the form of residues of division by rather small coprime modules allows one to avoid interbit carries and to considerably decrease numbers being processed. Moreover, owing to their natural internal parallelism, RNSs are brought forward in recent years as the most foreground basis for advanced high-performance computer technologies, in particular, such as multiprocessor [9], supercomputer, neural network-based [10], etc. ones.

THEORETICAL BASES OF RNS

The fundamental basis for RNSs is number theory, in particular, the Chinese remainder theorem [11]. Any integer positive number N in the decimal number system is represented in RNS in the form of residues $(b_1, b_2, \ldots, b_k)_{p_1, p_2, \ldots, p_k}$ of division of N by each of pairwise coprime modules $N_{10} = (b_1, b_2, \ldots, b_k)_{p_1, p_2, \ldots, p_k}$, where $b_i = N \bmod p_i$ and k is the number of modules. In this case, the condition $N \le P - 1$ $\left(P = \prod_{i=1}^{k} p_i\right)$ must be fulfilled.

The inverse transformation from the Krestenson basis into the decimal number system is rather cumbersome and is based on the use of the Chinese remainder theorem [12]

$$N = \left(\sum_{i=1}^{k} b_i B_i \right) \bmod P, \tag{1}$$

where $B_i = M_i m_i$, $M_i = P / p_i$, m_i is found from the expression $(M_i m_i) \bmod p_i = 1$, and the condition $\left(\sum_{i=1}^{k} B_i \right) \bmod P = 1$ must be fulfilled.

At present, the following three methods of searching for an inverse element are well known: (1) sequential exhaustive search for m_i until the condition $M_i m_i \bmod p_i = 1$ is satisfied; (2) with the help of the Euler function $m_i = M_i^{-1} \bmod p_i = M_i^{\varphi(p_i)-1} \bmod p_i$; (3) with the help of an extended Euclidean algorithm.

All of them are rather cumbersome, require large computational expenditures and time resources in executing divisions with residues, exponentiation, and finding of Euler's function. At the same time, all operations must be executed over very large numbers, which can lead to register overflows. The absence of the possibility of an analytical definition of an inverse element can also be considered as a drawback.

Ya. M. Nykolaychuk proposed the perfect form of RNS (PF RNS) in which the selection of modules is such that $M_i \bmod p_i = 1$, i.e., $m_i = 1$ [13]. This theory is further developed in [14, 15]. A method for choosing a system of modules for the PF RNS is shown and its modification is developed in which $m_i = \pm 1$. However, in these cases, processor registers whose number, as a rule, is equal to a power of two are not quite rationally used.

Proceeding from the aforesaid, the objective of this publication is the development of methods of selection of modules for the efficient use of processor registers and also the derivation of an analytical formula for searching for an inverse element for an appropriate selection of modules.