# Binomial Numeral Systems: Description and Applications to Numeration Problems

Alexei A. BORISENKO*, Nataliya I. KALASHNYKOVA**, Vyacheslav V. KALASHNIKOV***,   Junzo WATADA****, and Denys V. GUTENKO*

*Department of Computing and Industrial Electronics, Sumy State University, Sumy, Ukraine

**Faculty of Physics and Mathematics, Autonomous University of Nuevo León, Monterrey, Mexico; and Sumy State University, Sumy, Ukraine

***Department of Systems and Industrial Engineering, ITESM, Monterrey, Mexico; and CEMI RAS, Moscow, Russian Federation; and Sumy State University, Sumy, Ukraine

****The Graduate School of Information,. Production and Systems, Waseda University, Kitakyushu, 808-0135 Japan

.

**Abstract:** *We develop a new class of positional numeral systems, namely the binomial ones, which form a subclass of generalized positional numeral systems (GPNS). The binomial systems have wide range of applications in the information transmission, processing, and storage due to their error-detection capabilities. In this paper, the binomial numeral systems are well-defined, their prefix and compactness properties are established. Algorithms of generating binomial coding words (non-uniform and uniform) are presented, as well as an enhanced procedure of construction of constant weight Boolean combinations based upon the non-uniform binomial coding words. The correctness of this procedure is also established.*

    ***Keywords***     *Generalized positional numeral systems, Binomial numeral systems, Constant weight codes.*

## 1.Introduction

Positional numeral systems are widely used in computing. More complicated numeral systems, in which the register's weight need not be equal to the power of the system's base (like, for example, in the binary or decimal system), have not been thoroughly studied as yet. Such generalized positional numeral systems (GPNS) may have quite useful properties, like being noise-proof, easy in generating permutations, etc. (*see* [1]–[3]). These properties allow one to exploit the GPNS to develop specialized digital devices with high computational speed, reliability, and very low size and weight parameters. Moreover, the GPNS may serve as a base for: (*a*) generation of codes and construction of coding devices for the thorough error-control when processing, transmitting, and storing information; (*b*) development of algorithms and devices used when information is compressed and/or coded; (*c*) efficient solution of combinatorial optimization problems.

When combinatorial objects are generated and numerated, researchers use to develop special methods for each individual problem, which can be characterized as a principal drawback of such an approach (*cf*. [4], [5]). Therefore, a universal algorithm solving these problems at both the theoretical and practical levels would be very helpful. We propose a possible solution method based upon the GPNS. In particular, in this paper, a binomial numeral system is considered, which generates combinatorial objects making use of constant weight codes (*cf*. [6]). The total number of coding words in such codes is determined by binomial coefficients (*cf*. [7]).

The generalized positional numeral systems (GPNS) allow one to develop efficient algorithms and specialized digital devices (based upon these algorithms) due to the similarity of their structures. Thus the device cost is saved, and the high computational speed is attained (due to the hardware implementation, up to ten times higher compared to the universal computers). Moreover, as the GPNS are noise-proof, their digital devices use to be much more reliable and easy in trouble-shooting.

***ITESM, 2501 Av. Eugenio Garza Sada Sur
Monterrey 64849 Mexico
Phone: +52-81-8358-2000 ext. 5548
Fax:+52-81-8358-0771
e-mail: kalash@itesm.mx

It is worthwhile to develop digital devices using a GPNS and completing mainly logical and the simplest arithmetical operations with integer numbers, because these operations are realized by the GPNS in the most efficient way. Certain parts of such specialized devices, e.g. noise-proof counters, registers, etc., are of interest for the universal computers as well [8], [9].

To cope with the problem of noise-proof storage and transmission of information, a lot of various codes, both error-detecting and error-correcting, have been developed. Among those codes, it is worthy to distinguish the codes that detect errors not only during the information transmission and storage but also while it is processed. This class also includes the codes based upon the GPNS, whose strong sides are: (*i*) the simplicity of algorithms and devices for detecting errors, (*ii*) the structural regularity, (*iii*) the possibility to regulate the code's redundancy and hence its error-detecting capability depending upon the channel's adaptability. Such codes are quite applicable in specialized automatic controlling systems, as the information's downloading, processing, transmission, and development of controlling actions are all based upon the same GPNS code.

One of the important problems arising while storing and transmitting information is its compression, like for example by the optimal coding based on the Shannon-Fano and Huffman codes [10], [11]. Nowadays, the coding theory can boast with a quite wide arsenal of other ways to compress information, which however cannot exclude the development of new methods and/or improvement of the existing ones. One of those is the numeration of messages, which has the following advantages: (*a*) an algorithmic coding structure allowing an easy implementation, and (*b*) no need in a dictionary. Application of a GPNS permits one to expand the class of numerated messages and thus improve and simplify the algorithmic and technical realization of the information compression.

Both the numeration and de-numeration processes based upon the GPNS can be efficiently used to code the information. Thus we can obtain noise-proof codes of high stability and with simple keys used for the information security.

Finally, problems of combinatorial optimization are of special importance. In the most general form, these problems may even not have an objective function but stated in some preference terms. Such problems are usually solved by an exhaustive search, or when it is impossible, by random search procedures [2]. In both cases, the GPNS can provide many efficient ways of generating the combinatorial objects in order to find a path to an optimal solution.

Therefore, the generalized positional numeral systems (GPNS) propose a unified approach allowing one to solve efficiently a series of practical problems of various natures. As an example of such a GPNS, our paper presents a binary binomial numeral system. The latter is characterized with the use of binomial coefficients as weights of the binary digits (*cf.* [12]–[14]).

The rest of the paper is arranged as follows. In Section 2, we define the principal structure of the binomial calculus system. Section 3 presents the mains results establishing the key properties of the binomial systems, namely, the prefix and the compactness properties. Finally, Section 4 deals with the algorithms to generate and numerate binomial combinations of various lengths (non-uniform codes), the constant length (uniform codes), and the constant weight combinations. Conclusion, acknowledgement and the reference list complete the paper.

## 2. Binomial Systems

Now we describe one of the GPNS, namely the binomial system with the binomial weights and the binary alphabet $\{0,1\}$ (*cf.* [12]–[14]).

In a *k*-binomial system with *n* registers ($k < n$), the quantitative equivalent $QA_i$ of a code combination

$$A_i = \left( a_{j-1}, a_{j-2}, \ldots, a_0 \right), \quad i = 0, 1, \ldots, P-1 \qquad , \qquad \text{with}$$

$P = C_n^k$, where $j = j(i)$ is the combination's length, is defined as follows:

$$QA_i = a_{j-1}C_{n-1}^{k-q_j} + \ldots + a_\ell C_{n-j+\ell}^{k-q_{\ell+1}} + \ldots + a_0 C_{n-j}^{k-q_1}, \quad (1)$$

where the following conditions must hold: either

$$\begin{cases} q_0 = k, \\ j < n, \\ a_0 = 1, \end{cases} \qquad (2)$$

or

$$\begin{cases} n - k = j - q_0, \\ q_0 < k, \\ a_0 = 0. \end{cases} \qquad (3)$$

Here $q_0$ is the quantity of units (ones) in the binomial

number; $P$ is the system's range; $j$ is the quantity of registers in the binomial number (its length); $\ell = 0, 1, \ldots, j-1$ is the register's ordinal number; $q_\ell$ is the sum of the digits in the registers $j - 1$ through $\ell$, inclusively:

$$q_\ell = \sum_{i=\ell}^{j-1} a_i, \qquad (4)$$

with $q_j = 0$.

A positional numeral system must be finite, effective, and well-defined. However, it is not enough for a generalized positional system. In addition, it has to be a prefix code system, i.e. with the "prefix property": there is no valid code word in the system that is a prefix (start) of any other valid code word in the set. With a prefix code, a receiver can identify each word without requiring a special marker between words. The generalized positional numeral system should be also continuous, which means that for any number s from the system's range (except for the maximal number), there exists a combination, whose quantitative equivalent is equal to (s+1). All these properties of the binomial numeral systems will be established in the next section.

## 3. Binomial System Is Finite, Effective, Prefix, and Well-Defined

Formula (1) shows that the binomial numeral system is finite and effective, because there exists a numeration algorithm, which, after a finite of number of steps, converts the coding combination $A_i$ into its quantitative equivalent $QA_i$. Now the following theorem establishes the prefix property of the binomial numeral system. Its proof can be found in [12].

**Theorem 1** [12]. *The k-binomial numeral system with n registers (where k < n) is a prefix code system.* ∎

To show that the binomial system is well-defined, that is, two distinct coding combinations cannot be equivalent to the same numerical value, we proved the following result (again, *see* [12]).

**Theorem 2** [12]. *The k-binomial system with n registers (where k < n) is well-defined.* ∎

Theorems 1 and 2 have the following important corollary, which proves the compactness of the binomial numeral systems. For its proof, see also [12].

**Corollary 1** [12]. *The k-binomial system with n registers (k < n) is compact, that is, its range is complete and covers all the integers between 0 and ($C_n^k - 1$).* ∎

**Remark 1**. It is straightforward that for the $k$-binomial calculus system with $n$ registers, the range parameter $P$ is equal to $C_n^k$. ∎

## 4. Algorithms Generating Binomial Combinations

Table 4.1 contains the binomial combinations and their quantitative equivalents for the *k*-binomial system with *n* registers, where $n = 6$ and $k = 4$. They are generated by the following algorithm:

**Step 1**. An initial combination $A_0$ consisting of ($n - k$) zeros is composed and referred to as a *keyword*.

**Step 2.** The digit 1 is put into the right end register, and zero is added to the right side of it.

**Step 3.** Step 2 is repeated while the number of 1's in the coding word is less than $k$–1. If the number of 1's is equal to $k$–1, then go to Step 4.

**Step 4.** If the right end position contains zero, we replace it with 1. Go to Step 5.

**Step 5.** Check the number of 1's in the coding combination: if it equals $k$ but the 1's do not occupy the first $k$ registers counted from left to right, go to Step 6. Otherwise, *i.e.,* if the 1's occupy the first $k$ registers counted from left to right, then **STOP**: all the combination have been generated.

**Step 6.** Update the keyword $A_0$ by putting 1 as a prefix before the beginning of the keyword (i.e., its left end). If the total number of 1's in the keyword is less than $k$, go to Step 2.

**Tab. 4.1. Binomial coding combinations of various lengths (non-uniform code)**

| Binomial word | Its numerical equivalent |
|---|---|
| 00 | $0\, C_5^4 + 0\, C_4^4 = 0$ |
| 010 | $0\, C_5^4 + 1\, C_4^4 + 0\, C_3^3 = 1$ |
| 0110 | $0\, C_5^4 + 1\, C_4^4 + 1\, C_3^3 + 0\, C_2^2 = 2$ |
| 01110 | $0\, C_5^4 + 1\, C_4^4 + 1\, C_3^3 + 1\, C_2^2 + 0\, C_1^1 = 3$ |

| 01111 | $0\,C_5^4 + 1\,C_4^4 + 1\,C_3^3 + 1\,C_2^2 + 1\,C_1^1 = 4$ |
|-------|---------------------------------------------------------------|
| 100 | $1\,C_5^4 + 0\,C_4^3 + 0\,C_3^3 = 5$ |
| 1010 | $1\,C_5^4 + 0\,C_4^3 + 1\,C_3^3 + 0\,C_2^2 = 6$ |
| 10110 | $1\,C_5^4 + 0\,C_4^3 + 1\,C_3^3 + 1\,C_2^2 + 0\,C_1^1 = 7$ |
| 10111 | $1\,C_5^4 + 0\,C_4^3 + 1\,C_3^3 + 1\,C_2^2 + 1\,C_1^1 = 8$ |
| 1100 | $1\,C_5^4 + 1\,C_4^3 + 0\,C_3^2 + 0\,C_2^2 = 9$ |
| 11010 | $1\,C_5^4 + 0\,C_4^3 + 0\,C_3^2 + 1\,C_2^2 + 0\,C_1^1 = 10$ |
| 11011 | $1\,C_5^4 + 1\,C_4^3 + 0\,C_3^2 + 1\,C_2^2 + 1\,C_1^1 = 11$ |
| 11100 | $1\,C_5^4 + 1\,C_4^3 + 1\,C_3^2 + 0\,C_2^1 + 0\,C_1^1 = 12$ |
| 11101 | $1\,C_5^4 + 1\,C_4^3 + 1\,C_3^2 + 0\,C_2^1 + 1\,C_1^1 = 13$ |
| 1111 | $1\,C_5^4 + 1\,C_4^3 + 1\,C_3^2 + 1\,C_2^1 = 14$ |

The binomial systems find various important applications, in which the following useful features are exploited: (*i*) the binomial systems are noise-proof in the information transmission, processing, and storage; (*ii*) they are able to search, generate and numerate coding combinations with a constant weight; (*iii*) they can be used to construct noise-proof digital devices.

To detect errors with the aid of binomial coding combinations, they should be completed with zeros to obtain uniform (*n*–1)-digital binomial coding words given in Table 4.2.

**Tab. 4.2**. **Binomial coding combinations of the same length (uniform code)**

| NN | Binomial word | Binomial uniform word |
|----|---------------|------------------------|
| 0 | 00 | 00000 |
| 1 | 010 | 01000 |
| 2 | 0110 | 01100 |
| 3 | 01110 | 01110 |
| 4 | 01111 | 01111 |
| 5 | 100 | 10000 |
| 6 | 1010 | 10100 |
| 7 | 10110 | 10110 |
| 8 | 10111 | 10111 |
| 9 | 1100 | 11000 |
| 10 | 11010 | 11010 |
| 11 | 11011 | 11011 |
| 12 | 11100 | 11100 |
| 13 | 11101 | 11101 |
| 14 | 1111 | 11110 |

The main tokens of errors in a binomial coding combination is either the number of 1's being greater than *k*, or the number of zeros exceeding $n-k$. The principal feature of the binomial noise-proof code is its ability to detect errors while processing information. This feature allows one to arrange the throughout control in the information processing channels involving the digital devices.

### 4.1. Generation of binomial coding combinations with a constant weight

Next, Table 4.3 shows transformation of binomial coding combinations to coding words with constant weight: this is done by adding (to the right end) either 1's if the binomial combination contains $n-k$ zeros, or adding zeros if the combination comprises *k* ones, until the combination's lengths reaches *n*.

**Tab. 4.3. Binomial coding combinations of the constant weight**

| NN | Binomial word | Binomial constant weight word |
|----|---------------|-------------------------------|
| 0 | 00 | 001111 |
| 1 | 010 | 010111 |
| 2 | 0110 | 011011 |
| 3 | 01110 | 011101 |
| 4 | 01111 | 011110 |
| 5 | 100 | 100111 |
| 6 | 1010 | 101011 |
| 7 | 10110 | 101101 |
| 8 | 10111 | 101110 |
| 9 | 1100 | 110011 |
| 10 | 11010 | 110101 |
| 11 | 11011 | 110110 |

| 12 | 11100 | 111001 |
|----|-------|--------|
| 13 | 11101 | 111010 |
| 14 | 1111  | 111100 |

Each binomial combination (column 2 of Table 4.3) has the corresponding combination with the constant weight (column 3 of Table 4.3), hence the former is a compressed image of the latter. If one needs to label a combination with the constant weight by some traditional numeral system number (e.g., decimals of column 1 in Table 4.3), formula (1) has to be used. In the latter case, a compression of binomial numbers is completed.

Algorithms of search and generation of binomial combinations and those with constant weights can be also found in [14]. Now we describe one of modifications of such algorithms and prove its efficiency as follows. This method is based upon the fact that the range of binomial numbers of length n and with parameter $k$ ($k < n$) coincides with the range of the constant weight coding combinations with $k$ units among n registers. Therefore, the formal description of the algorithm is as follows:

**Step 1.** Select an arbitrary non-uniform binomial coding combination.

**Step 2.** If the coding combination ends with the digit 1, then put zeros into all registers up to the right end (register $n$), which is considered as auxiliary. The thus obtained combination ending with 0 will be the combination with the constant weight.

**Step 3.** If the coding combination ends with the digit 0, then set units (ones) into all registers up to the right end (register $n$, or the auxiliary register). The thus created combination ending with 1 will be the combination with the constant weight.

**Step 4.** Verify that the thus obtained combination is indeed with the constant weight by counting the total number of ones (units). If this number is $k$ then the combination is indeed a desired one. Select another non-uniform binomial coding combination and go to Step 2. If all the non-uniform binomial coding combinations have been already selected, then **STOP**: all the constant weight combinations of this range have been generated.

The above algorithm generates the complete range of the corresponding combinations of the constant weight, which is confirmed by the following theorem.

**Theorem 3**. *With the aid of the above algorithm, for every non-uniform binomial combination of length n with parameter k (k < n), one obtains the unique corresponding coding combination with (the constant) weight k and length n.* ∎

In order to illustrate some properties of the above-described algorithm, let us evaluate its computational cost (running time). If we denote by $\tau > 0$ a time unit necessary to complete one operation (comparison, introduction of an extra register (digit), and increasing the binomial word's length by one unit), then it can be demonstrated that the computational time needed to transform a non-uniform $(n,k)$-binomial coding combination with length $r$ into the corresponding coding combination with (the constant) weight $k$ and length $n$, equals

$$T = 3\tau(n-r) + 2\tau. \qquad (5)$$

Since non-uniform $(n,k)$-binomial coding combination may have various length parameters, then the running time of the algorithm in question will be different, even for fixed values of $n$ and $k$. Now we estimate the average running time $T_{ave}$ of the algorithm applied to all possible non-uniform $(n,k)$-binomial coding combinations. In order to do that, we will make use of the average length $r_{ave}$ of the non-uniform $(n,k)$-binomial coding combinations, which, according to [15], is equal to

$$r_{ave} = \frac{k(n-k)(n+2)}{(k+1)(n-k+1)}. \qquad (6)$$

Substituting the average length (6) in the running time estimate (5), we come to the average running time needed for transformation of a non-uniform $(n,k)$-binomial coding combination into the corresponding constant weight $k$ and length $n$ word:

$$T_{ave} = 3\tau\left[n - \frac{k(n-k)(n+2)}{(k+1)(n-k+1)}\right] + 2\tau. \qquad (7)$$

As an example of such a transformation presented above in Table 4.3, we can calculate the running time needed to transform each of the 15 non-uniform $(6,4)$-binomial combination, by formula (5); *see* Table 4.4 below:

**Tab. 4.4. Running time needed to transform (6,4)-binomial words into the constant weight words**

| NN | Binomial word | Binomial constant weight | Running time |
|----|---------------|--------------------------|--------------|

| | | word | |
|---|---|---|---|
| 0 | 00 | 001111 | $14\tau$ |
| 1 | 010 | 010111 | $11\tau$ |
| 2 | 0110 | 011011 | $8\tau$ |
| 3 | 01110 | 011101 | $5\tau$ |
| 4 | 01111 | 011110 | $11\tau$ |
| 5 | 100 | 100111 | $8\tau$ |
| 6 | 1010 | 101011 | $5\tau$ |
| 7 | 10110 | 101101 | $5\tau$ |
| 8 | 10111 | 101110 | $5\tau$ |
| 9 | 1100 | 110011 | $5\tau$ |
| 10 | 11010 | 110101 | $5\tau$ |
| 11 | 11011 | 110110 | $5\tau$ |
| 12 | 11100 | 111001 | $5\tau$ |
| 13 | 11101 | 111010 | $8\tau$ |
| 14 | 1111 | 111100 | $8\tau$ |

Making use of formula (7) we find the average time needed to transform a non-uniform (6,4)-binomial combination into a (constant) weight 4 coding word as follows:

$$T_{ave} = 3\tau\left[6 - \frac{4(6-4)(6+2)}{(4+1)(6-4+1)}\right] + 2\tau = 7.2\tau.$$

## 5. Conclusion

In this paper, we have described the error-detecting binomial numeral systems capable of transmitting, processing and storing information. The systems can also generate and numerate combinatorial configurations, like, for example, coding words with a constant weight, as well as compositions, combinations with repetitions, etc. Moreover, the binomial systems can be applied to produce efficient information compression and defense. The latter is the goal of our further research.

### References
[1] Stakhov, A.P.: "Golden Section Codes", *Radio and Communication*, Moscow, 1984 (*in Russian*).

[2] Stoyan, Yu.G., and Sokolovsky, V.Z.: "Solution of Some Extremum Problems by Method of Contracting Neighborhoods", *Naukova Dumka*, Kiev, 1980 (*in Russian*).

[3] Borisenko, A.A.: "Numerical coding based upon combinatorial calculus systems", Proceedings of the International Conference on Theoretical and Applied Aspects of Program Systems Development (TAAPSD'2010), Kiev, October 4–8, pp. 98-104, 2010 (*in Russian*).

[4] Amelkin, V.A.: "Numerical Coding Methods", Nauka, Novosibirsk, 1986 (*in Russian*).

[5] Reingold, E.M., Nievergelt, J., and Deo, N.: "Combinatorial Algorithms: Theory and Practice", Prentice Hall, Englewood Cliffs, New Jersey, 1977.

[6] Borisenko, A.A., and Kulik, I.A.: "Binomial Coding", Sumy State University, Sumy, Ukraine, 2010 (*in Russian*).

[7] Anderson, J.A.: "Discrete Mathematics with Combinatorics", Prentice Hall, Upper Saddle River, New Jersey, 2001.

[8] Borisenko, A.A.: "Binomial Calculus and Counters", Sumy State University, Sumy, Ukraine, 2008 (*in Russian*).

[9] Oberman, R.M.M.: "Counting and Counters", Wiley & Sons, Hoboken, New Jersey, 1981.

[10] Tsymbal, V.P.: "Information and Coding Theory", Vyshaya Shkola, Kiev, 1977 (*in Russian*).

[11] Kuzmin, I.V., and Kedrus, V.A.: "A Basic Theory of Information and Coding", Vyshaya Shkola, Kiev, 1977 (*in Russian*).

[12] Borisenko, A.A., Kalashnikov, V.V., and Kalashnykova, N.I.: "Binomial calculus: Advantages and prospects", ICIC Express Letters, Vol. 2, No. 2, pp. 123-130, 2008.

[13] Borisenko, A.A.: "Introduction to Theory of Binomial Calculus", Universitetskaya Kniga, Sumy, Ukraine, 2004 (*in Russian*).

[14] Borisenko, A.A.: "Binomial Calculus: Theory and Applications", Universitetskaya Kniga, Sumy, Ukraine, 2004 (*in Russian*).

[15] Kulik, I.A.: "On the average length of binary binomial numbers", Visnik Sumskogo Derzhavnogo

Universitetu, Vol. 12 (70), pp. 106-112, 2004 (*in Russian*).

[16] Borisenko A.A.,Gutenko D.V. "Generation of equilibrium codes based on uneven binomial numbers",Visnik Sumskogo Derzhavnogo Universitetu, No. 3, Vol. 2, pp. 164-170, 2010 (*in Ukrainian*).

**Alexei A. BORISENKO**
He received the M.E degree in radio electronics from Kharkov Institute for Radio and Electronics, Kharkov, Ukraine, in 1970, and his Ph.D. degree in Computing in 1976. He received the Dr. Sc. (Habilitation) Degree in Information Systems and Computing in 1991 in Kharkov. He has worked at the Kharkov Institute for Radio and Electronics since 1976, and at Sumy State University, Sumy, Ukraine, since 1980. Now he chairs the Department of Electronics and Computing at Sumy University. His present research interests include the application of factorial and binomial numeral systems to the codification, information transmission and compressing problems.

**Nataliya I. KALASHNYKOVA**
She received the M.S. degree in applied mathematics from Novosibirsk State University, Novosibirsk, Russia, in 1978, and the Dr. degree in Operations Research in 1989 from Institute of Mathematics of the Siberian Division of the USSR Academy of Sciences in Novosibirsk. She has worked as Associate Prof. of Dept. of Mathematical Economics Methods at Sumy State University, Sumy, Ukraine, and at Faculty of Physics and Mathematics of Autonomous University of Nuevo León (UANL), Monterrey, Mexico. Her present research interests include development of conjectural variations equilibrium (CVE) theory and applications to electricity markets. She is a member of National System of Researchers (SNI) and Society of Mathematics of Mexico (SMM).

**Vyacheslav V. KALASHNIKOV**
He received the M.S. degree in pure and applied mathematics from Novosibirsk State University, Novosibirsk, Russia, in 1977, and the Ph.D. degree in Operations Research from Institute of Mathematics of the Siberian Division of the USSR Academy of Sciences in Novosibirsk, in 1981. He received the Dr. Sc. (Habilitation) Degree in 1995 from the Central Economics and Mathematics Institute of the Russian Academy of Sciences, Moscow. He has worked as a Prof. of Dept. Mathematical Economics Methods at Sumy State University, Sumy, Ukraine, and at ITESM, Campus Monterrey, Mexico. His present research interests include development of the theory and applications of Variational Inequality and Complementarity problems. He is a member of the National System of Researchers (SNI) in Mexico, American Mathematical Society (AMS), Society for Industrial and Applied Mathematics (SIAM).

**Junzo WATADA**
Received the B.Sc. and M.Sc. degrees in electrical engineering from Osaka City University, Osaka, Japan, and the Ph.D. degree from Osaka Prefecture University, Osaka. Currently, he is a Professor of Management Engineering, Knowledge Engineering and Soft Computing at the Graduate School of Information, Production Systems, Waseda University, Fukuoka, Japan. His professional interests include soft computing, tracking system, knowledge engineering, and management engineering. Prof. Watada is a recipient of the Henri Coanda Medal Award from Invention in Romania in 2002. He is a Life Fellow of Japan Society for Fuzzy Theory and intelligent Informatics (SOFT). He is the Principal Editor, a Co-Editor, and an Associate Editor of various international journals, including the International Journal of Intelligent Decision Technology, ICIC Express Letters, and the International Journal of Systems and Control Engineering, Fuzzy Optimization Decision Making.

**Denys V. GUTENKO**
He recieved the M.S. degree in Electronic Systems from Sumy State University, Sumy, Ukraine, in 2008. Now he continues his study at the Ph.D. courses at Sumy State University (Dept. of Electronics and Computing) with his major at Informational Technologies (IT).