DOI: 10.17323/2587-814X.2020.2.84.92

About the possibility of determining the prefix and suffix of a word by subwords of fixed length

Galina N. Zhukova[a]
E-mail: galinanzhukova@gmail.com

Yuri G. Smetanin[b]
E-mail: smetanin.iury2011@yandex.ru

Mikhail V. Ulyanov[c]
E-mail: muljanov@mail.ru

[a] National Research University Higher School of Economics
Address: 20, Myasnitskaya Street, Moscow 101000, Russia

[b] Federal Research Center "Computer Science and Control", Russian Academy of Sciences
Address: 40, Vavilova Street, Moscow 119333, Russia

[c] Trapeznikov Institute of Control Sciences, Russian Academy of Sciences
Address: 65, Profsoyuznaya Street, Moscow 117997, Russia

Abstract

In applied problems of business informatics related to data analysis (in particular, in the analysis and forecasting of time series, in the study of log files of business processes, etc.), problems of qualitative analysis arise. Qualitative analysis methods often use symbolic coding as a way of presenting information about the processes under study. In a number of situations, due to the fragmentation of such descriptions, the problem arises of reconstructing a complete symbolic description of a process (word) from its successive fragments (subwords). From the multiset of all subwords of a sufficiently large length, the original word is uniquely restored. In the case of insufficiently long subwords, several different reconstructions of the original word are possible. The number of feasible reconstructions can be reduced by determining the suffix and prefix of the reconstructed word. A method is proposed for determining the prefix and suffix of a word consisting of $k-1$ symbols each on the basis of multiset V of subwords of a fixed length equal to k. We accept the hypothesis that this multiset is generated by a window of a fixed length k of one symbol shift in an unknown word. The method for determining the

prefix and suffix is based on the construction and analysis of the matrix formed by subwords from V written in rows in arbitrary order and the use of the operator acting on multisets of characters of the alphabet formed by neighboring columns of this matrix. The method is capable of determining the prefix $a_1a_2 \ldots a_{k-1}$ and suffix $b_1b_2 \ldots b_{k-1}$, if $a_i \neq b_i$ for any i from 1 to $k-1$. If in the prefix and suffix $a_i \neq b_i$ only for some values of i, the characters in the corresponding positions are determined, and $a_j = b_j$ for the remaining characters. In the worst case, the method concludes that $a_i = b_i$ for any i from 1 to $k-1$, but does not determine the characters themselves. This is a situation in which the prefix and suffix coincide but cannot be determined.

Key words: word reconstruction; prefix; suffix; multiset of subwords; subwords of fixed length; shift operator.

Citation: Zhukova G.N., Smetanin Yu.G., Ulyanov M.Yu. (2020) About the possibility of determining the prefix and suffix of a word by subwords of fixed length. *Business Informatics*, vol. 14, no 2, pp. 84–92. DOI: 10.17323/2587-814X.2020.2.84.92

Introduction

In the applied areas of business informatics related to data analysis, such as time series analysis and forecasting [1–6], research of business process log files [7], etc., problems of qualitative analysis arise. In this case, one of the commonly used methods for presenting information about processes is symbolic encoding [8]. Furthermore, a description of the behavior of a time series or a business process is encoded with a word over a finite alphabet which is the object of further research. However, in a number of cases, including the analysis of business processes and time series, researchers do not know the whole word, but a multiset of subwords that are consecutive fragments of a certain word. Since in this case the positions of the subwords in the original word are unknown, the problem of reconstruction arises, i.e. the restoration of the unknown word from the original set of subwords [9–17]. This problem relates to a special section of discrete mathematics, namely the combinatorics on words [18]. The objects of research in the combinatorics on words are words over arbitrary alphabets, and the subject of research is the study of the combinatorial properties of various sets of words, both finite and infinite. In real-life applied problems, information about words is often incomplete; for example, such a situation is inevitable in the analysis of infinite time series measured over finite time intervals.

We note that one of the important areas of practical application of the methods of combinatorics on words is the field of bio-molecular models and processes. At the same time, work with fragmentary information is characteristic of a number of bio-informatics and genomics problems. For example, the problem of sequencing genomes [19, 20] is essentially the problem of reconstructing words under conditions of strong restrictions, implying unique reconstruction.

The problems of reconstructing words over a finite alphabet have different statements, differing both in the amount of information available and in the restrictions on feasible solutions [21–23]. Usually, these problems, as problems with incomplete information, are complex, and obtaining any additional information obviously allows us to reduce the set of solutions under consideration.

In a qualitative analysis of time series [24, 25], the coding of the observed variable can be carried out in a certain alphabet, for example, (A, B, C, D, E, F), which symbols can be used to name half-segments of the observed values of

the variable in the ascending order. For example, A is the name of the half segment of the smallest value, F is the largest. Since observations are recorded in discrete time, the description of the values of the time series by the names of half-segments is a word over the alphabet of names. If the observed process is characterized by sharp outliers of the observed value (up to level F) relative to the basal level (A, B) in one moment, as well as sharp drops (from F to B), then the resulting codewords of time series will not contain subwords CDE and EDC. In this case, if the initial data are subwords (scattered fragments of observations), then the problem of reconstructing a word from subwords is the problem of restoring the entire description of a time series under the assumption of the peculiarities of its behavior.

A similar situation arises when reconstructing business process log files in the presence of fragmented information. When describing business processes by the graph theory apparatus [7], a model (business process graph) can be represented as follows: process states are encoded by named vertices, and state transitions are encoded by edges identified with stages of the business process. Then the record of a particular implementation of a business process is a word over the alphabet of vertex names that reflects the state transition order. If the process is physically distributed between various organizations and executors, then most likely we will receive information about its complete performance in the form of a set of subwords. In addition, prohibited subwords can be interpreted as violations of the model (the regulation of the business process). The arising reconstruction problem, without forbidden subwords, means the possibility of a complete reconstruction of the entire process corresponding to the theoretical model.

Thus, it is of interest to study in detail the various versions of the word reconstruction problem with a certain set of subwords of shorter length, interpreted as a set of consecutive fragments of an unknown word. Moreover, of interest are both the case when the reconstructed word does not contain a predetermined forbidden subword and the case with the presence of forbidden subwords. One of the possible solutions to this problem, based on subwords of fixed length, in the shift by one symbol hypothesis, was proposed by the authors in [26, 27]. However, the set of possible reconstructions can be large and the problem arises of a possible reduction in the number of feasible solutions for the "correct" reconstructed word. We want to obtain additional information from the initial set of subwords, which will be useful in reducing the resulting set of reconstructions. We are talking about the possibility of restoring and / or determining the pattern of the prefix and suffix of an unknown word, which, as part of the reduction procedure, will lead to the consideration of only those words that have the obtained patterns of prefix and suffix. It is the problem that is the subject of this article.

1. Terminology and notation

Further in the text of the article, the following notation will be used:

$\Sigma = \{s_1, s_2, \ldots s_l\}$ – alphabet where s_i is i-th symbol of the alphabet;

Σ^k – the k-fold Cartesian product (Cartesian product of set Σ, i.e. the set of k-element tuples);

$\Sigma^* = \bigcup_{k=0}^{\infty} \Sigma^k$ – transitive closure of Σ (the set of all possible tuples);

w – a word (above the alphabet), which is a sequence of characters of the alphabet, while the actual characters of the alphabet are words by definition;

$L(\cdot) : L(C) = W$ where $C \subseteq \Sigma^*$ is a set of tuples, W is a set of words. Operator $L(\cdot)$ is an operator acting on a set of tuples; $L(\cdot)$ creates a set of words consisting of characters from Σ;

a_i is i-th character of word w, $a_i \in \Sigma$;

$w = a_1 a_2 \ldots a_n \in L(\Sigma^n)$ is an arbitrary word consisting of n characters of alphabet Σ;

$|w| = n$ – word length, defined as the number of its elements;

$L_k = L(\Sigma^k) = \{w \mid |w| = k\}$ – the set of all words of length k over alphabet Σ.

Let $w = a_1 a_2 \ldots a_n \in L(\Sigma^n)$, and $k < n$, then $v = a_{i_1} a_{i_2} \ldots a_{i_k}, 1 \le i_1, i_2 = i_1 + 1, i_k = i_{k-1} + 1 \le n -$ a subword of a word w of length k;

$Q(w, i, k)$ is an operator that gives the subword of length k of word w, starting with a character in position i.

Let $|w| = n$, then the operator is defined for $i + k - 1 \le n$ so that

$$Q(a_1 a_2 \ldots a_n, i, k) = a_i a_{i+1} \ldots a_{i+k-1},$$

$$Q(w, i, k) \in L_k;$$

For the following two operators, we assume that $|w| = n \ge 2$ and $1 \le k < n$:

$P(w, k) = Q(w, 1, k) = a_1 a_2 \ldots a_k \in L_k$ is the prefix of length k of word w;

$S(w, k) = Q(w, n - k + 1, k) = a_{n-k+1} \ldots a_n \in L_k$ is the suffix of length k of word w;

$SH1(w, k)$ is a shift by one operator. The operator, defined when $|w| > k$, generates a set of subwords of length k (the cardinality of this set is $|w| > k + 1$), performing a shift of a window of length k along word w, starting from the leftmost position of word w:

$$SH1(w, k) = \{v_j \mid j = 1, |w| - k + 1; v_j = Q(w, i, k)\}.$$

2. Statement of the problem

Afterwards, we consider as a given: the length of the subword is k, the number of subwords is m, and the original multiset V of subwords over alphabet Σ, considered as the basis for the reconstruction of some unknown word w:

$$V = \left\{v_i \mid i = \overline{1, m}; v_i = a_{i1} a_{i2} \ldots a_{ik} \in L_k\right\}.$$

The hypothesis of shift one accepted by the authors states that V is a multiset of subwords of shift by one symbol alongside some unknown word w, where $|w| = n = m + k - 1$ and

$$V = SH1(w, k) = \\ = \left\{v_j \mid j = 1, n - k + 1; v_j = Q(w, j, k)\right\}.$$

Informal statement: Under the hypothesis of shift one, is it possible to determine the prefix and suffix of length $k - 1$ of the unknown word w, or to obtain any meaningful information about its prefix and suffix using only multiset V?

Mathematical statement: For a given multiset V with the length k of the subwords and the number of subwords equal to m, determine prefix $P(w, k - 1)$ and suffix $S(w, k - 1)$ of length $k - 1$ of the original word $w = a_1 a_2 \ldots a_n$, and indicate the conditions under which a solution is possible.

3. Method for determining the prefix and suffix

First, we note that the main problem, both in the aspect of the reconstruction problem and in the aspect of the problem of determining the suffix and prefix, is that we were initially given a multiset of subwords V, but not a tuple of subwords. The main difficulty is connected with the loss of order in the original subwords obtained by the shift operator.

We begin the solution of this problem by constructing matrix A consisting of m rows and k columns whose rows are words v_i from set V. Words from set V can be represented in form $v_i = a_{i1}, a_{i2}, \ldots a_{ik}$, and the elements of matrix A are the symbols of alphabet Σ, i.e. $A = (a_{ij})$, where a_{ij} is a symbol of the alphabet at the j-th position in the i-th word of multiset V in the order in which they are listed.

We explicitly write matrix A in the direct sequence of the window of shift by one symbol. Obviously, in reality, in the order of enumeration in multiset V, we will observe some permutation of words of the direct sequence, and, consequently, the corresponding permutation of the rows of matrix A:

$$A = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_{m-1} \\ v_m \end{pmatrix} = \begin{pmatrix} a_1, a_2, ..., a_k \\ a_2, a_3, ..., a_{k+1} \\ a_3, a_4, ..., a_{k+2} \\ \vdots \\ a_{n-k}, a_{n-k+1}, ..., a_{n-1} \\ a_{n-k+1}, a_{n-k+2}, ..., a_n \end{pmatrix}.$$

The solution to the problem of determining the prefix and suffix is based on the analysis of neighboring columns of this matrix. Let us consider the first and second columns. In each of them, at any permutation of rows, there will be symbol a_2 that is the second symbol of the unknown word w, and symbol a_3 that is the third symbol of w, etc. If the matching pairs of characters are deleted from these two columns, only a_1 and a_{n-k+2} remain if they are not equal. If they are different, we get their exact values. If a_1 and a_{n-k+2} coincide, then all characters in these columns will be crossed out, and we will get information that unknown, but coincident characters are in the corresponding positions of the prefix and suffix. We continue such an analysis for all $k-1$ pairs of neighboring columns of matrix A. Provided that after crossing out pairs of matching characters we always have a mismatched pair, we will restore the prefix and suffix of length $k-1$ of the unknown word w.

We describe the method formally.

We introduce a tuple of all symbols of the alphabet for which multiplicities of elements are allowed

$$C = \left(s_1^{(\alpha 1)}, s_2^{(\alpha 2)}, ..., s_l^{(\alpha l)}\right),$$

where multiplicity 0 yields an empty set $s_i^{(0)} = \varnothing$ in this position. We define operator G acting on the i-th column of matrix A and creating tuple C_i containing, for all characters of the alphabet, their multiplicity in accordance with the number of characters in this column

$$GC(A, i) = C_i = \left(s_1^{(\alpha 1)}, s_2^{(\alpha 2)}, ..., s_l^{(\alpha l)}\right).$$

We apply operator G to two columns of matrix A, and denote:

$$GC(A, i) = C_i = \left(s_1^{(\alpha 1)}, s_2^{(\alpha 2)}, ..., s_l^{(\alpha l)}\right),$$

$$GC(A, k) = C_k = \left(s_1^{(\beta 1)}, s_2^{(\beta 2)}, ..., s_l^{(\beta l)}\right).$$

We introduce operator GS of obtaining a character that acts on two tuples of columns of matrix A according to the following rule:

$$GS(A,i,k) = \begin{cases} \bigcup_{j=1}^{l} s_j^{(\alpha_j - \beta_j)}, \left(s_j^{(\alpha_j)} \in GC(A,i)\right), \\ \qquad s_j^{(\beta_j)} \in GC(a,k), \\ s_j^{(\alpha j - \beta_j)} = \varnothing, \text{если } a_j - \beta_j \le 0. \end{cases}$$

Now apply operator GS to two consecutive columns of matrix A. Due to the structure of successive columns of matrix A described above, the result of action of operator GS will be either a symbol or an empty set. Note that if $GS(A, i, i+1) \neq \varnothing$, then $GS(A, i+1, i) \neq \varnothing$ too. In this case, we define the $n-k+i$-th prefix character $a_i = GS(A, i, i+1)$ and the $n-k+i$-th character $a_{n-k+i} = GS(A, i+1, i)$ of the unknown word, which is the i-th character of suffix of length $k-1$.

For example, if $GS(A, 1, 2) = s_i$, then we know the first character $a_1 = s_i$ of the unknown word w (the first character of the prefix) and, in this situation, the value $GS(A, 2, 1)$ is not necessarily an empty set. Let $GS(A, 2, 1) = s_j$, and we get the first character $a_{n-k+2} = s_j$ of the suffix. If $GS(A, 1, 2) \neq \varnothing$, then, it is obvious that $GS(A, 2, 1) \neq \varnothing$ too and we get information that $a_1 = a_{n-k+2}$, but at the same time the symbol of the alphabet itself at these positions remains unknown to us.

Since we have $k-1$ consecutive pairs of columns, then if for each consecutive pair of columns operator GS gives a non-empty set, then using the "+" operation to indicate the concatenation of characters, we get the solution:

$$P(w, k-1) = a_1 a_2 \ldots a_{k-1} = \sum_{i=1}^{k-1} GS(A, i, i+1),$$

$$S(w, k-1) = a_{n-k+2} \ldots a_n = \sum_{i=1}^{k-1} GS(A, i+1, i).$$

If for each pair operator GS yields an empty set, then the prefix and suffix characters remain unknown, but at the same time, we get information about their equality as subwords

$$P(w, k-1) = S(w, k-1).$$

In a general case, we get information about prefix and suffix characters in the form of some pattern, and if these are specific characters, then they are located at the same positions of the prefix and suffix, and if the characters cannot be determined, then we have information about that at these positions the prefix and suffix characters match.

Let us give an example for word $w = abbaaabb$ in alphabet $\Sigma = \{a, b\}$ and the set of subwords obtained by the shift to one symbol operator with a window of width three. In this case, $k=3$, $m=6$, $n=8$, and matrix A has the form:

$$A = \begin{pmatrix} abb \\ bba \\ baa \\ aaa \\ aab \\ abb \end{pmatrix}.$$

Applying operator G to the three columns of matrix A gives the following tuples:

$$GC(A, 1) = C_1 = (a^{(4)}, b^{(2)}),$$

$$GC(A, 2) = C_2 = (a^{(3)}, b^{(3)}),$$

$$GC(A, 3) = C_3 = (a^{(3)}, b^{(3)}).$$

and we get $GC(A, 1, 2) = a$, $GC(A, 2, 1) = b$, and $GC(A, 2, 3) = GC(A, 3, 2) = \varnothing$. Thereby, we get the prefix pattern $P(w, 2) = a^*$ of length two of word $w = abbaaabb$, and the suffix pattern $S(w, 2) = b^*$, where symbol $*$ denotes an unknown but matching symbol in the corresponding positions of the prefix and suffix (in fact, this is the symbol "b").

4. Application to the reconstruction problem

In one of the previous articles [26], the authors proposed a solution to the problem of complete reconstruction, under the conditions of a given multiset of subwords and one shift hypothesis. In some cases, the number of reconstructions determined by the number of Euler paths or cycles in the corresponding de Brain multi-graph can be significant [26].

Let us introduce the set of possible word reconstructions by the initial set V:

$$W = \left\{(w) \;\middle|\; |w| = m,\ k-1 = n,\ V = SH1(w, k)\right\},$$

In this case, if $|W| \geq 2$, then reconstruction is possible and there can be many of them. Let w^* be the word under consideration, that is unknown to us, based on which the set V is obtained, where $V = SH1(w^*, k)$. Then when choosing a possible reconstruction in set W, we select only those words that possess the prefix and suffix obtained by the operator GS, taking into account patterns with possibly unknown characters. As a result we obtain

$$\tilde{W} = \left\{ \begin{array}{l} (w) \;\middle|\; P(w, k-1) = \sum_{i=1}^{k-1} GS(A, i, i+1), \\ S(w, k-1) = \sum_{i=1}^{k-1} GS(A, i+1, i) \end{array} \right\},$$

where $w^* \in \tilde{W}$ is guaranteed.

This leads to a reduction in the resulting set of reconstructions, since we consider only those words that have the given prefix and suf-

fix patterns. Moreover, this approach can be applied not only to reduce a finite set of reconstructions, but to consider the prefix as a pattern for choosing the initial arcs for the Euler paths in the de Brain multi-graph when constructing the reconstruction [26].

Conclusion

In this article, in the aspect of solving the problem of reconstruction of symbolic descriptions of time series and logs of business processes, a solution to the problem of determining the prefix and suffix of an unknown word is proposed. The solution is based on the assumption that the full set of subwords of fixed length k, originally generated by the window of length $k$going alongside an unknown word with a shift to one symbol, is initially given. A solution has been obtained that allows us to acquire information about the prefix and suffix of an unknown word or some patterns for the prefix and suffix. The proposed solution allows us to obtain additional information about possible reconstructions, and thereby reduce the number of possible word reconstructions for a given set of subwords. In the best case, the proposed method allows us to determine the prefix and suffix of length k of an unknown word, or, in the worst case, to state that the prefix and suffix coincide.

The results can be used in conjunction with solving the reconstruction problem [26, 27] to reduce the set of possible reconstructions during qualitative analysis in such problems of business informatics as analysis of time series and logs of business processes. ■

Acknowledgments

The study reported here was funded by RFBR, project number 19-07-00150.

References

1. Ding H., Trajcevski G., Scheuermann P., Wang X., Keogh E. (2008) Querying and mining of time series data: Experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, vol. 1, no 2, pp. 1542–1552. DOI: 10.14778/1454159.1454226.
2. Kurbalija V., Radovanović M., Geler Z., Ivanović M. (2011) The influence of global constraints on DTW and LCS similarity measures for time-series databases. *Advances in Intelligent and Soft Computing*, vol. 101, pp. 67–74. DOI: 10.1007/978-3-642-23163-6_10.
3. Wu Y.-L., Agrawal D., el Abbadi A. (2000) A comparison of DFT and DWT based similarity search in time-series databases. Proceedings of the *Ninth International Conference on Information and Knowledge Management (CIKM '00), McLean, VA, 6–11 November 2000*, pp. 488–495.
4. Bemdt D.J., Clifford J. (1994) Using dynamic time warping to find patterns in time series. *AAAI-94 Workshop on Knowledge Discovery in Databases*, pp. 359–370. Available at: https://www.aaai.org/Papers/Workshops/1994/WS-94-03/WS94-03-031.pdf (accessed 15 March 2020).
5. Dreyer W., Dittrich A.K., Schmidt D. (1994) Research perspectives for time series management systems. *SIGMOD Record*, vol. 23, no 1, pp. 10–15.
6. Keogh E.J., Pazzani M.J. (1998) An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback. Proceedings of the *Fourth International Conference on Knowledge Discovery and Data Mining (KDD'98), New York, 27–31 August 1998*, pp. 239–241.
7. Andersen B. (1999) *Business processes improvement toolbox*. New York: ASQ Quality Press.
8. Lin J., Keogh E., Wei L., Lonardi S. (2007) Experiencing SAX: A novel symbolic representation of time series. *Data Mining and Knowledge Discovery*, vol. 15, no 2, pp. 107–144. DOI: 10.1007/s10618-007-0064-z.
9. Acharya J., Das H., Milenkovic O., Orlitsky A., Pan S. (2014) String reconstruction from substring compositions. *SIAM Journal on Discrete Mathematics*, vol. 29, no 3, pp. 1340–1371.
10. Manvel B., Meyerowitz A., Schwenk A., Smith K., Stockmeyer P. (1991) Reconstruction of sequences. *Discrete Mathematics*, vol. 94, no 3, pp. 209–219. DOI: 10.1016/0012-365X(91)90026-X.

11. Carpi A., de Luca A. (2001) Words and special factors. *Theoretical Computer Science*, vol. 259, no 1–2, pp. 145–182.

12. de Luca A. (1999) On the combinatorics of finite words. *Theoretical Computer Science*, vol. 218, no 1, pp. 13–39.

13. Dudík M., Schulman L.J. (2003) Reconstruction from subsequences. *Journal of Combinatorial Theory. Series A*, vol. 103, no 2, pp. 337–348. DOI: 10.1016/S0097-3165(03)00103-1.

14. Erdős P.L., Ligeti P., Sziklai P., Torney D.C. (2006) Subwords in reverse-complement order. *Annals of Combinatorics*, vol. 10, no 4, pp. 415–430. DOI: 10.1007/s00026-006-0297-3.

15. Fici G., Mignosi F., Restivo A., Sciortino M. (2006) Word assembly through minimal forbidden words. *Theoretical Computer Science*, vol. 359, no 1–3, pp. 214–230. DOI: 10.1016/j.tcs.2006.

16. Levenshtein V.I. (2001) Efficient reconstruction of sequences from their subsequences or supersequences. *Journal of Combinatorial Theory, Series* A, Vol. 93, pp. 310–332.

17. Piña C., Uzcátegui C. (2008) Reconstruction of a word from a multiset of its factors. *Theoretical Computer Science*, vol. 400, no 1–3, pp. 70–83. DOI: 10.1016/j.tcs.2008.01.052.

18. Lothaire M. (2002) *Algebraic combinatorics on words.* Cambridge, UK: Cambridge University Press.

19. Gusfield D. (1997) *Algorithms on strings, trees, and sequences: Computer science and computational biology.* Cambridge, UK: Cambridge University Press.

20. Skiena S.S., Sundaram G. (1995) Reconstructing strings from substrings. *Journal of Computational Biology*, vol. 2, no 2, pp. 333–353.

21. Leont'ev V.K., Smetanin Y.G. (2002) Problems of Information on the set of words. *Journal of Mathematical Sciences*, vol. 108, no 1, pp. 49–70. DOI: 10.1023/A:1012705332306.

22. Levenshtein V.I. (1997) Restoring objects based on the minimum number of distorted samples. *Doklady Akademii Nauk*, vol. 354, no 5, pp. 593–596 (in Russian).

23. Krasikov I., Roditty Y. (1997) Note: On a reconstruction problem for sequences. *Journal of Combinatorial Theory, Series A*, no 77, pp. 344–348.

24. Ulyanov M.V., Smetanin Yu.G. (2013) Determining the characteristics of Kolmogorov complexity of time series: An approach based on symbolic descriptions. *Business Informatics*, no 2, pp. 49–54 (in Russian).

25. Smetanin Yu.G., Ulyanov M.V. (2014) Measure of symbolical diversity: Combinatorics on words as an approach to identify generalized characteristics of time series. *Business Informatics*, no 3, pp. 40–46 (in Russian).

26. Smetanin Yu.G., Ulyanov M.V. (2014) Reconstruction of a word from a finite set of its subwords under the unit Shift hypothesis. I. Reconstruction without for bidden words. *Cybernetics and Systems Analysis*, vol. 50, no 1, pp. 148–156.

27. Smetanin Yu.G., Ulyanov M.V. (2015) Reconstruction of a word from a finite set of its subwords under the unit Shift hypothesis. II. Reconstruction with forbidden words. *Cybernetics and Systems Analysis*, vol. 51, no 1, pp. 157–164. DOI: 10.1007/s10559-015-9708-y.

About the authors

Galina N. Zhukova

Cand. Sci. (Phys.-Math.);

Associate Professor, School of Software Engineering, Faculty of Computer Science,
National Research University Higher School of Economics,
20, Myasnitskaya Street, Moscow 101000, Russia;

E-mail: galinanzhukova@gmail.com

ORCID: 0000-0003-1835-7422

Yuri G. Smetanin

Dr. Sci. (Phys.-Math.);

Chief Researcher, Federal Research Center "Computer Science and Control",
Russian Academy of Sciences,
40, Vavilova Street, Moscow 119333, Russia;

E-mail: smetanin.iury2011@yandex.ru

ORCID: 0000-0003-0242-6972

Mikhail V. Ulyanov

Dr. Sci. (Tech.);

Leading Researcher, Laboratory of Scheduling Theory and Discrete Optimization,
V.A. Trapeznikov Institute of Control Sciences, Russian Academy of Sciences,
65, Profsoyuznaya Street, Moscow 117997, Russia;

E-mail: muljanov@mail.ru

ORCID: 0000-0002-5784-9836