УДК: 519.24
DOI: 10.25559/SITITO.15.201902.431-440

Вероятностная модель шумов для периодических символьных последовательностей

Г. Н. Жукова[1], Ю. Г. Сметанин[2], М. В. Ульянов[3,4*]
[1] Национальный исследовательский университет «Высшая школа экономики», г. Москва, Россия
101000, Россия, г. Москва, ул. Мясницкая, д. 20
[2] Федеральный исследовательский центр «Информатика и управление» Российской академии наук, г. Москва, Россия
119333, Россия, г. Москва, ул. Вавилова, д. 44, корп. 2
[3] Институт проблем управления им. В.А. Трапезникова Российской академии наук, г. Москва, Россия
117997, Россия, г. Москва, ул. Профсоюзная, д. 65
[4] Московский государственный университет имени М.В. Ломоносова, г. Москва, Россия
119991, Россия, г. Москва, ГСП-1, Ленинские горы, д. 1
[*] muljanov@mail.ru

Аннотация

В целях анализа методов поиска циклов и выявления их особенностей и чувствительности к шумам различных типов необходимо моделирование шумов с заданными характеристиками. Для построения почти периодических символьных последовательностей предложены две вероятностные модели шума. Модели позволяют вносить в периодическую последовательность различные типы шума, такие как изменение, добавление и удаление символов. Таким образом на основе периодической символьной последовательности получается почти периодическая последовательность.

Необходимо обеспечить заданный уровень шума в построенной почти периодической последовательности. Требуемый уровень вносимого шума гарантирует двухуровневая модель, в которой на первом уровне на основе выбранной исследователем дискретной случайной величины определяются позиции для внесения шума, а на втором уровне с использованием случайной величины, моделирующей собственно шум, в соответствующие позиции вносится необходимое изменение. Вторая модель основана на внесении шума с вероятностью, зависящей от уровня шума, в каждом элементе последовательности. Наблюдаемый уровень шума рассчитывается на основе расстояния Левенштейна между исходной периодической последовательностью и построенной зашумленной. Наблюдаемый уровень шума всегда несколько меньше уровня вносимого шума, поскольку при вычислении расстояния Левенштейна может быть найден более короткий путь получения зашумленной последовательности из периодической, чем использованный при построении почти периодической последовательности.

Проведено сравнение предложенных моделей по близости обеспечиваемого ими уровня шума к заданному уровню шума. Вычислительный эксперимент показал, что наблюдаемый уровень шума ближе к заданному для двухуровневой модели. Программная реализация данных моделей будет использована в дальнейшем для исследования алгоритмов поиска циклов в зашумленных периодических символьных последовательностях.

Ключевые слова: символьная последовательность, периодическая последовательность, последовательность с шумами, вероятностная модель шума.

Финансирование: данная работа была подготовлена при финансовой поддержке Российского фонда фундаментальных исследований в рамках научных проектов № 19-07-00150 «Методы построения и анализа информативных символьных представлений по фрагментарным данным» и № 19-07-00151 «Комплексные научные и экспериментальные междисциплинарные исследования по моделированию поведения разливов нефти на море под воздействием гидрометеорологических условий».

Для цитирования: *Жукова Г. Н., Сметанин Ю. Г., Ульянов М. В.* Вероятностная модель шумов для периодических символьных последовательностей // Современные информационные технологии и ИТ-образование. 2019. Т. 15, № 2. С. 431-440. DOI: 10.25559/SITITO.15.201902.431-440

A Stochastic Model of Noises for Periodic Symbol Sequences

G. N. Zhukova[a], Yu. G. Smetanin[b], M. V. Ulyanov[c,d*]
[a] National Research University Higher School of Economics, Moscow, Russia
20 Myasnitskaya St., Moscow 101000, Russia
[b] Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, Russia
44-2 Vavilova St., Moscow 119333, Russia
[c] V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences, Moscow, Russia
65 Profsoyuznaya St., Moscow 117997, Russia
[d] Lomonosov Moscow State University, Moscow, Russia
1, Leninskie gory, Moscow 119991, Russia
[*] muljanov@mail.ru

Abstract
The construction of almost periodic sequences is needed for the analysis of the cycles' detection methods, identification of symbolic sequences' features and sensitivity analysis. Two probabilistic models of noise are proposed for constructing almost periodic symbolic sequences. Models provide various types of noise in a periodic sequence, such as changing, adding and deleting characters. Thus, on the basis of a periodic symbolic sequence, an almost periodic sequence is constructed.
It is necessary to ensure a given level of noise in the constructed almost periodic sequence. The required level of simulated noise is guaranteed by a two-level model, in which the positions for simulated noise are determined at the first level, based on the discrete random variable, chosen by the researcher. At the second level, the necessary changes are made at the corresponding positions using a random variable that simulates the noise itself. The second model is based on simulated noise with probability (depending on the noise level) in each element of the sequence. The observed noise level is calculated based on the Levenshtein distance between the original periodic sequence and the almost periodic sequence that we constructed.
The observed noise level is always somewhat less than the level of the noise simulated, since when calculating the Levenshtein distance, a shorter way of obtaining a noisy sequence from a periodic one can be found than the one that is used in constructing an almost periodic sequence.
A comparison of the proposed models for the proximity of the noise level they provide to a given noise level is made. The computational experiment showed that the observed noise level is closer to that provided by the two-level model. The software implementation of these models will be used in future to study the algorithms for finding cycles in noisy periodic symbolic sequences.

Keywords: symbolic sequence, periodic sequence, sequence with noise, probabilistic model of noise.

Funding: This work was prepared with the financial support of the Russian Foundation for Basic Research in the framework of scientific projects No. 19-07-00150 "Methods for Constructing and Analyzing Informative Symbolic Representations Based on Fragmentary Data" and No. 19-07-00151 "Complex Scientific and Experimental Interdisciplinary Studies of Modeling the Behavior of Oil Spills at Sea Under The Influence of Hydrometeorological Conditions."

For citation: Zhukova G.N., Smetanin Yu.G., Ulyanov M.V. A Stochastic Model of Noises for Periodic Symbol Sequences. *Sovremennye informacionnye tehnologii i IT-obrazovanie* = Modern Information Technologies and IT-Education. 2019; 15(2):431-440. DOI: 10.25559/SITITO.15.201902.431-440

1. Введение

В широком смысле объектами исследования в данной статье являются периодические процессы. Периодичность, то есть повторяемость наблюдаемых явлений через определенные промежутки времени, играет важную роль во многих областях, как в повседневной деятельности, так и в научных исследованиях. В качестве очевидных примеров можно привести колебательные процессы в физике, движение астрономических объектов, биоритмы, экономические циклы. Фундаментальной научной задачей является разработка методов анализа и прогнозирования периодических последовательностей, описывающих процессы произвольной природы, в условиях неполных и искаженных данных [1-9]. Во многих прикладных областях искажения и неполнота данных приводят к фрагментарности и потере части информации, следствием которой является утрата цикличности. В реальных задачах анализа циклических процессов входные данные оказываются почти всегда искаженными шумами. Природа этих шумов различна, как различны и последствия искажений для разных прикладных задач [10-15]. Очевидно, возникает задача устранения искажений и восстановление данных в неискаженном виде. Наиболее широко в литературе, естественно, представлено очень много методов борьбы с шумами и искажениями в изображениях и сигналах [16, 17].

Благодаря циклическому характеру большого ряда процессов и наличию случайных искажений, вызванных внешними факторами, для адекватного описания таких процессов подходит математическая модель символьной последовательности с зашумленными циклами. Описание входной информации в терминах слов над фиксированным алфавитом позволяет упростить анализ исследуемых свойств. При символьном кодировании символы соответствуют состояниям анализируемой системы по квантам времени и возникает лишенная посторонних деталей модель системы, дающая возможность исследовать такие свойства, как периодичность, среднее время пребывания в определенных состояниях и т. д.

При анализе периодичности следует заранее определить порог точности: насколько малые изменения значений измеряемых параметров допустимы для того, чтобы считать данные повторяющимися. После определения этого порога можно перейти к символьному кодированию, при котором диапазон возможных значений параметра разбивается на конечное число интервалов, каждый из которых обозначается своим символом из некоторого конечного алфавита. Символьное кодирование позволяет не только уменьшить вычислительную сложность анализа, но и устраняет постороннюю информацию, которая не нужна для анализа периодичности и отвлекает внимание от действительно важных характеристик. Это особенно полезно при анализе больших данных. Например, в задачах, где достаточна лишь бинарная характеризация, высокая точность числовых представлений признаков приводит к неоправданно большим объемам информации и трудоемким вычислениям без повышения качества анализа по сравнению с представлением в двоичном алфавите.

Символьное кодирование искаженных циклических временных рядов приводит к появлению зашумленных периодических символьных последовательностей. Под шумами будем понимать искажения данных, заключающиеся либо в замене символов алфавита на некоторых заранее неизвестных позициях другими символами, либо выпадения или вставки символов. Искажения первого вида встречаются чаще. Они могут быть вызваны как сбоями в системе сбора и передачи данных, так и характером самих изучаемых процессов. Например, к этому виду относятся помехи в радиосигналах, помехи на видеоизображениях, погодные колебания и изменения средней температуры, колебания продажи сезонных товаров и т. д. Выпадения и вставки символов чаще всего связаны именно со сбоями оборудования, но не всегда: например, при выборе информации из баз данных можно столкнуться с ситуацией пропусков данных. При анализе ДНК могут возникать искажения обоих указанных типов.

Для исследования таких зашумленных периодических последовательностей с целью выявления информативных характеристик очевидный интерес представляет задача поиска циклов, то есть подпоследовательностей с длиной, равной периоду. В настоящее время предложено достаточно много методов поиска циклов в таких последовательностях, основанных на различных математических моделях [18-22].

В целях анализа этих методов и выявления их особенностей и чувствительности к шумам различных типов необходимо моделирование шумов с заданными характеристиками, что и является предметом исследования в данной статье, которая посвящена вероятностным моделям шума в периодических символьных последовательностях. Модели позволяют учитывать особенности анализируемых временных рядов и допустимых шумов и использовать эти особенности в качестве априорной информации для повышения качества анализа.

На основе представленных моделей можно анализировать уровень шума исходного циклического процесса, при условии, что возможный цикл известен или получен каким-либо способом. В случае, если имеется несколько возможных вариантов цикла, возможно выбрать среди них такой, что уровень шума почти периодической последовательности будет минимален.

2. Цели исследования

Для исследования методов поиска циклов в зашумленных символьных последовательностях и определения их чувствительности к шумам различных типов необходимо моделирование символьных шумов с заданными характеристиками. В этих целях необходимо генерировать синтетические тестовые данные – зашумленные символьные последовательности с заранее известными свойствами. Такая генерация возможна на основе некоторой модели шумов, в частности — вероятностной. В связи с этим основной целью исследования является разработка таких вероятностных моделей символьного шума, вносимого в чисто периодическую символьную последовательность, которые позволили бы:

- оперировать единообразно с шумами различных типов;
- обеспечить заданный уровень символьного шума раздельно по типам шумов;
- единообразно генерировать простые и комбинированные шумы;
- дать исследователю возможность гибко варьировать позиции шума в исходной чисто периодической последовательности.

3. Типы шумов

Перевод временного ряда из цифрового представления в символьное осуществляется методами символьного кодирования [23]. Анализ особенностей методов символьного кодирования позволяет для циклических временных рядов с шумами разделить шумы, вносимые в чисто периодические символьные последовательности на следующие три типа:

Шум замены. Такой шум возникает в ситуации, когда при кодировании некоторого периода в исходном временном ряде значение отсчета попадает в другой сегмент, кодируемый другим символом, по сравнению с отсчетом, отстоящим на период ранее. И, хотя ожидаемым является близлежащий сегмент, мы можем наблюдать как резкие всплески значений, так и резкие падения, приводящие к замене символа, который стоял в этой позиции в предыдущем периоде, на любой символ кодирующего алфавита. Отметим, что шум этого типа не приводит к изменению длины периода.

Шум добавления. Возникновение такого типа шума, приводящего к изменению длины периода, связано с тем, что исходный процесс, подвергаемый символьному кодированию, возвращается на некотором периоде в условное «начальное» состояние позже, чем в предыдущем. Такое различие длин может характеризоваться добавлением не обязательно одного символа, а быть может и нескольких, сильно искажая длину периода.

Шум удаления. По аналогии с предыдущим типом шума, шум удаления возникает при условии, что исходный процесс, подвергаемый символьному кодированию, возвращается на некотором периоде в условное «начальное» состояние раньше, чем в предыдущем периоде.

Для детального анализа алгоритмов идентификации циклов в зашумленных периодических символьных последовательностях интерес представляет исследование способности этих методов определять цикличность в зависимости от уровня шума в каждом из описанных типов.

4. Определение уровня шума

Для экспериментального определения уровня шума мы фиксируем некоторый алфавит символов и рассматриваем строки (последовательности) над этим алфавитом. В нашем случае требуется измерить расстояние между чисто периодической последовательностью и циклической последовательностью с шумами. Существует несколько различных способов формализации понятия расстояния между строками, один из них, наиболее общий и простой, носит название редакционного расстояния. Термин и метод предложен В.И. Левенштейном в [24] и достаточно часто редакционное расстояние называется расстоянием Левенштейна. Идея основана на преобразовании одной строки в другую с помощью фиксированных операций редактирования — вставка символа в первую строку, удаление символа из первой строки, замена символа в первой строке и «не операция» над правильным символом. Последовательность этих операций называется редакционным предписанием. По определению редакционное расстояние между двумя строками определяется как минимальное число необходимых редакционных операций — вставок, удалений и замен, необходимых для преобразования первой строки во вторую. При этом совпадения символов не являются операциями, которые учитываются в редакционном расстоянии. Если строки совпадают, то их редакционное расстояние равно нулю для любой длины строк. Расстояние Левенштейна обычно вычисляется методом динамического программирования на основании теоремы, доказательство которой можно найти, например, в [25].

Поскольку редакционные операции, фигурирующие в расстоянии Левенштейна, совпадают с введенными типами шумов, то суммарное число внесенных единичных (односимвольных) шумов в периодическую последовательность, содержащую несколько циклов, будет верхней оценкой расстояния Левенштейна.

В рамках данной статьи нас интересует тот факт, насколько зашумлена вся последовательность, поэтому под уровнем шума данного типа будем понимать в дальнейшем отношение числа единичных зашумлений к общей длине исходной периодической последовательности. Альтернативой является вычисление среднего уровня шума в цикле – в этом случае нужно общее число единичных зашумлений делить на число циклов (периодов) в идеальной (исходной) последовательности.

Поскольку для дальнейших исследований методов идентификации важно обеспечить заранее заданный уровень шума, то вероятностная модель, ориентированная на внесение шумов, должна обеспечивать соответствующий уровень символьного шума. Уровень внесенного шума, вообще говоря, не будет равен уровню шума, вычисленному с помощью расстояния Левенштейна, поскольку может найтись более короткий путь получения конкретной зашумленной последовательности из периодической, чем тот, что был использован при построении почти периодической последовательности. Другими словами, расстояние Левенштейна может быть строго меньше числа внесенных единичных шумов. Связь уровней внесенного и наблюдаемого (по расстоянию Левенштейна) шумов будет исследована далее.

5. Двухуровневая вероятностная модель символьных шумов. Содержательное описание.

Создание синтетической зашумленной периодической символьной последовательности начинается с генерации чисто периодической последовательности. При этом задается длина цикла (период) и число циклов в последовательности. Далее с помощью равномерного случайного генератора создается слово над фиксированным алфавитом с заданной длиной — периодом, после чего это слово повторяется заданное число раз. Мы получаем чисто периодическую последовательность, которая является заготовкой для создания зашумленной последовательности на основе некоторой вероятностной модели.

Для решения задачи создания зашумленной последовательности в качестве основной модели авторы предлагают следующую двухуровневую вероятностную модель. В этой модели верхний уровень отвечает за позиции, в которые будет вноситься шум, а нижний уровень определяет собственно символьный шум в данной позиции. Исключение составляет шум удаления, для которого, очевидно, модель нижнего уровня не нужна.

Предлагаемая модель верхнего уровня представляет собой некоторую дискретную случайную величину, выбранную исследователем, с областью определения либо на длине исходного цикла, либо на полной длине идеальной периодической по-

следовательности. Далее рассматривается случай дискретной случайной величины, определенной на множестве номеров всех элементов периодической последовательности. Вначале производится многократное разыгрывание этой случайной величины, получается множество различных номеров элементов последовательности, в которых будет внесен шум. Если очередной полученный при разыгрывании случайной величины номер равен какому-либо из ранее выпавших номеров, то происходит дополнительное разыгрывание случайной величины до тех пор, пока не будет получено нужное количество разных номеров.

На нижнем уровне модели происходит разыгрывание случайной величины со значениями 0, 1 и 2, что соответствует шумам замены, добавления и удаления. Вероятности, с которыми эта случайная величина принимает свои значения, пропорциональны заданным уровням шума соответствующего типа.

Рассмотрим более детально процесс внесения шумов для каждого из введенных типов.

Для каждой полученной позиции, в которой требуется внести шум замены, выполняется разыгрывание случайной величины нижнего уровня, которая определяет номер заменяемого символа в алфавите (отличный от исходного символа). Такая замена может добавить не более единицы в расстояние Левенштейна между исходной и зашумленной последовательностями. Детали о связи числа изменений символов исходной последовательности и расстоянием Левенштейна будут представлены в описании экспериментальной части работы.

В случае шума добавления случайная величина нижнего уровня определяет добавляемый символ, которым может быть любой символ алфавита, в том числе такой же, как и символ в данной позиции. Новый символ включается в последовательность после исходного символа с выбранным на верхнем уровне номером.

Для моделирования шума удаления в строящуюся зашумленную последовательность не добавляется символ исходной периодической последовательности с номером, полученным на верхнем уровне, для которого случайная величина нижнего уровня, отвечающая за тип шума, приняла значение 2.

6. Математическое описание двухуровневой вероятностной модели шума

Пусть требуется получить псевдослучайную последовательность, содержащую n циклов длины m с уровнем шума замены α, вставки - β, удаления - γ. Под уровнем шума будем подразумевать отношение числа символов, обеспечивающих шум, к длине периодической последовательности. Так, уровень внесенного шума замены

$$\propto = \frac{r}{nm},$$

где r – число символов исходной последовательности, замененных на какие-то другие символы под влиянием случайных искажающих факторов. Уровень шума вставки и удаления определяется аналогично. Тогда число вставленных новых символов βnm, удаленных «по ошибке» символов γnm, всего получается $(1+\beta-\gamma)nm$ символов, а число символов исходной периодической последовательности, которые были заменены на другие символы алфавита при внесении шума, равно αnm. Поскольку число вставленных, удаленных и измененных символов должно быть целым числом, то значения уровня шума α, β и γ могут принимать только такие значения, чтобы при умножении α, β и γ на nm получалось целое число. Если считать уровень шума как отношение числа символов, обеспечивающих шум, к длине полученной почти периодической последовательности, получим

$$\tilde{\alpha} = \frac{\propto nm}{(1+\beta-\gamma)nm} = \frac{\propto}{(1+\beta-\gamma)}.$$

При этом относительное отличие уровня шума, вычисленное на основании длины исходной и полученной последовательностей, можно получить по формуле

$$\frac{\propto}{\tilde{\alpha}} = 1+\beta-\gamma,$$

из которой следует, что относительное отличие уровней шума, вычисленных по исходной или полученной последовательности, равно разности уровней шума замены и вставки. Аналогичная ситуация наблюдается и для остальных двух типов шума. Далее везде будем считать уровень шума как отношение числа символов, обеспечивающих шум, к длине исходной периодической последовательности, поскольку все преобразования замены и удаления производятся над символами только этой последовательности.

Для построения псевдослучайной последовательности с заданным уровнем шумов трех типов возьмем за основу периодическую последовательность, содержащую n циклов длины m, эта последовательность состоит из nm символов. На первом этапе (верхний уровень модели) получим с помощью генератора псевдослучайных чисел $(\propto+\beta+\gamma)nm$ различных целых чисел от 1 до nm, после чего для каждого из этих чисел разыгрывается случайная величина со значениями 0, 1 и 2, что соответствует типам шума (замена, вставка и удаление). При построении зашумленной последовательности символы с номерами, в которых не предполагается искажений, копируются из исходной периодической последовательности, символы с номерами, в которых производится удаление, пропускаются.

Для определения символа, на который нужно заменить символ, выбранный для замены на первом этапе, с помощью генератора псевдослучайных чисел получаем псевдослучайное число из равномерного дискретного распределения от 1 до $r-1$, где r – мощность алфавита. Все символы алфавита нумеруются от 1 до r, будем обозначать их $s_1, s_2, \supset, s_r$. Пусть на втором этапе получено псевдослучайное число k для символа s_q, подлежащего замене, тогда s_q заменяется на s_k, если $k<q$, иначе s_q заменяется на s_{k+1}, что обеспечивает равновероятность замены символа s_q на любой отличный от него символ алфавита.

Вставка символа осуществляется следующим образом. С помощью генератора псевдослучайных чисел получаем псевдослучайное число k из равномерного дискретного распределения от 1 до r, где r – мощность алфавита. Вставляем символ s_k после символа a_q, выбранного для вставки на первом этапе, т.е. в строящуюся почти периодическую последовательность добавляем и символ a_q, и дополнительный символ s_k. Обратим внимание на то, что номера элементов последовательности a_q, после которых вставляем новый символ, это номера из исходной последовательности.

После того, как на последнем этапе добавится примерно βnm

новых символов и будет удалено примерно γnm символов исходной последовательности, длина последовательности станет равна примерно $(1+\beta-\gamma)nm$, в ней будет содержаться около αnm измененных символов исходной последовательности и приблизительно $(1-\alpha-\gamma)nm$ исходных символов, оставшихся без изменений. В случае $\beta \neq \gamma$ длина полученной зашумленной почти периодической последовательности будет отличаться от длины исходной периодической последовательности.

Тот факт, что в построенной последовательности будет не ровно βnm новых символов, а примерно такое число, связан с тем, что после определения номеров символов, подлежащих внесению шума, для каждого из этих номеров символов разыгрывается случайная величина, определяющая тип шума. Таким образом, общий шум будет равен $\propto+\beta+\gamma$, но уровни шума каждого типа будут только близки к требуемым значениям. Если нужно, чтобы уровни внесенного шума каждого вида были равны требуемым значениям, можно разыгрывать случайную величину с равновероятными значениями от 1 до nm, до тех пор, пока не получится множество из $\propto nm$ различных чисел. При построении зашумленной последовательности после элементов исходной последовательности с этими номерами будет добавлен новый символ. Затем с помощью того же генератора псевдослучайных чисел получим набор из βnm различных чисел от 1 до nm, не совпадающих с числами из первого множества; элементы периодической последовательности с номерами из этого второго множества не включаются в строящуюся почти периодическую последовательность. Наконец, генерируется третье множество номеров, не входящих в первые два, такое что члены исходной последовательности заменяются на любые другие символы алфавита.

Описанная процедура гарантирует лишь внесение шума заданного уровня для каждого типа шума, но это вовсе не означает, что при определении расстояния Левенштейна будет сделано именно такое число изменений каждого типа, как при построении последовательности. Детальное исследование зависимости долей вставленных, удаленных и замененных символов исходной последовательности (при вычислении расстояния Левенштейна) от уровня внесенного шума каждого типа планируется провести в дальнейшем.

7. Вероятностная модель шума в каждом символе

Рассмотрим альтернативную модель получения псевдослучайной последовательности с шумами трех описанных выше типов. За основу берем периодическую последовательность, содержащую n циклов длины m с уровнем шума замены α, вставки - β, удаления – γ, причем $\propto+\beta+\gamma<1$ (заметим, что это условие автоматически выполняется при $\propto\leq 0.3$, $\beta\leq 0.3$, $\gamma\leq 0.3$). Только теперь под уровнем шума будем понимать вероятность проведения соответствующей операции (замены, вставки или удаления) над каждым символом последовательности.

Для каждого из mn символов последовательности с помощью генератора псевдослучайных чисел будем получать реализацию случайной величины Y с законом распределения, представленным в табл. 1а.

Таблица 1а. **Закон распределения случайной величины Y**
Table. 1a. **Distribution Law of the Random Value distribution law**

Y	0	1	2	3
p	$1-\propto-\beta-\gamma$	α	β	γ

Если случайная величина $Y=0$, то рассматриваемый символ остается без изменений, а при $Y=1$ заменяется на любой из оставшихся символов алфавита, вероятность выбора любого другого символа алфавита равна $P(s_i)=\frac{1}{r-1}$, где r – мощность алфавита. Если $Y=2$, то после имеющегося символа вставляется еще один символ алфавита, для чего равномерно случайно выбирается любой из символов алфавита, вероятность выбора каждого из символов алфавита равна $P(s_i)=\frac{1}{r}$. В случае $Y=3$ символ удаляется.

Отличием этой модели шума от двухуровневой является то, что не только наблюдаемый, но и вносимый уровень шума каждого типа будет достигаться лишь в среднем. Т.е. для каждой отдельной полученной последовательности и уровень общего шума, и уровни шумов каждого типа будут несколько отличаться от желаемого, поскольку выборочные частоты лишь близки к вероятностям, но обычно не равны им. Насколько значительное отличие заданного шума от полученного можно наблюдать рассмотрим далее.

8. Вычислительный эксперимент

Последовательность циклов из неповторяющихся символов.

Для сравнения желаемого уровня шума с полученным с помощью модели шума в каждом символе для серии периодических последовательностей было вычислено расстояние Левенштейна между исходной периодической последовательностью и соответствующей зашумленной. Были рассмотрены случаи бинарного алфавита, а также алфавиты, содержащие 3, 5, 10, 12, 15, 20 и 50 символов. Для каждого алфавита были сгенерированы циклы, содержащие все символы алфавита ровно по одному разу. Исходная периодическая последовательность состояла из m полных циклов, число m принимало значения в диапазоне от 4 до 40 с шагом 2. Такой способ построения исходной последовательности позволил избежать влияния структуры цикла на результат вычислительного эксперимента, поскольку структура всех используемых циклов одинакова – цикл содержит все допустимые значения ровно по одному разу. Любой такой цикл можно описать как последовательность целых чисел от 0 до $r-1$ (r – мощность алфавита), которая состоит из номеров соответствующих символов в алфавите. Любой цикл из всех символов алфавита, взятых ровно по одному разу, можно представить в таком виде за счет соответствующего упорядочивания символов алфавита.

Был рассмотрен желаемый уровень шума $\propto=\beta=\gamma$ с уровнем γ, равным 0,1, 0,2 и 0,3, при этом использовалась модель шума в каждом символе. Сравнение отношения расстояния Левенштейна между периодической и зашумленной последовательностями к длине исходной периодической последовательности с желаемым общим уровнем шума $\propto+\beta+\gamma$ показало, что для бинарного алфавита полученный уровень шума заметно ниже желаемого, с ростом мощности алфавита отличие желае-

мого и полученного уровней шума уменьшается. Полученный уровень шума не может быть больше желаемого (по построению), а меньше может быть, поскольку зашумленную последовательность иногда можно получить из соответствующей периодической за меньшее число редакционных предписаний, чем потребовалось при построении зашумленной последовательности. Кроме того, вносимый уровень шума каждого типа является случайной величиной, поэтому наблюдаемая частота того или иного редакционного предписания будет близка к соответствующей вероятности, но не будет ей равна.

В табл. 1- 3 приведены результаты вычислительного эксперимента при $\propto= \beta = \gamma = 0{,}1$. Приводятся значения числа полных циклов в исходной последовательности, минимальный, средний и максимальный шум, а также минимальный, средний и максимальный ожидаемый шум. Под шумом подразумевается отношение расстояния Левенштейна между зашумленной и исходной периодической последовательностью к длине периодической последовательности, а ожидаемый шум равен отношению суммы редакционных предписаний, выполненных при генерации зашумленной последовательности, к длине исходной последовательности. Поскольку оба вида шума являются случайными величинами, то для более точного анализа для каждого набора параметров (мощность алфавита и число полных циклов) было получено по 100 зашумленных последовательностей. Для каждой из них было вычислено общее число редакционных предписаний, выполненных при построении и расстояние Левенштейна до исходной периодической последовательности. По каждой выборке из 100 зашумленных последовательностей были вычислены характеристики, приведенные в табл. 1-3. В табл. 3 представлены для всех рассматриваемых мощностей алфавита диапазоны изменения всех характеристик в зависимости от числа полных циклов.

Таблица 1. **Сравнение желаемого и полученного уровней шума, бинарный алфавит**

Table 1. **Comparison of the desired and received noise levels, binary alphabet**

число циклов	минимальный шум	средний шум	максимальный шум	минимальный ожидаемый шум	средний ожидаемый шум	максимальный ожидаемый шум
4	0,00	0,24	0,50	0,00	0,31	0,75
6	0,08	0,24	0,50	0,08	0,31	0,67
8	0,06	0,24	0,44	0,06	0,31	0,63
10	0,10	0,22	0,40	0,10	0,28	0,55
12	0,08	0,23	0,38	0,13	0,30	0,54
20	0,10	0,22	0,35	0,15	0,29	0,50
30	0,12	0,23	0,33	0,17	0,30	0,45
38	0,14	0,24	0,33	0,21	0,31	0,43

Таблица 2. **Сравнение желаемого и полученного уровней шума, алфавит из пяти символов**

Table 2. **Comparison of desired and received noise levels, five-character alphabet**

число циклов	минимальный шум	средний шум	максимальный шум	минимальный ожидаемый шум	средний ожидаемый шум	максимальный ожидаемый шум
4	0,10	0,27	0,50	0,10	0,29	0,55
6	0,07	0,27	0,47	0,07	0,30	0,53
8	0,10	0,28	0,43	0,10	0,30	0,53
10	0,10	0,27	0,42	0,12	0,30	0,46
20	0,17	0,27	0,38	0,17	0,29	0,40
30	0,19	0,28	0,35	0,21	0,30	0,41
38	0,21	0,27	0,35	0,23	0,30	0,39

Таблица 3. **Сравнение желаемого и полученного уровней шума, алфавиты мощности от 2 до 50**

Table 3. **Comparison of the desired and received noise levels, power alphabets from 2 to 50**

мощность алфавита	минимальный шум	средний шум	максимальный шум	минимальный ожидаемый шум	средний ожидаемый шум	максимальный ожидаемый шум
2	0,00-0,14	0,22-0,24	0,33-0,50	0,00-0,21	0,28-0,31	0,43-0,75
3	0,06-0,18	0,26-0,29	0,33-0,50	0,06-0,20	0,29-0,34	0,39-0,75
5	0,07-0,21	0,27-0,28	0,33-0,50	0,07-0,23	0,23-0,30	0,39-0,55
10	0,15-0,24	0,27-0,29	0,33-0,50	0,15-0,25	0,29-0,31	0,35-0,52
15	0,17-0,25	0,28-0,29	0,32-0,43	0,17-0,27	0,30-0,31	0,35-0,47
20	0,18-0,25	0,28-0,29	0,32-0,43	0,18-0,27	0,29-0,31	0,33-0,44
50	0,21-0,27	0,28-0,29	0,31-0,37	0,21-0,28	0,30	0,33-0,40

Последовательность циклов произвольной структуры. Для алфавита из 5 символов были построены циклы длины 5, 10, 15, 25 и 50, каждый символ цикла был получен генератором псевдослучайных чисел с равномерным дискретным распределением, вероятность выбора каждого из 5 символов равна 0,2. Зашумленная последовательность строилась при уровне шума $\propto= \beta = \gamma = 0{,}1$. Во всех экспериментах средний ожидаемый шум был равен 0,3, а средний шум, вычисленный по расстоянию Левенштейна – примерно 0,265. Минимальные и максимальные наблюдаемые значения по выборкам объема 100 для каждого набора параметров (период и число циклов) приведены в табл. 4.

Таблица 4. **Минимальный и максимальный уровень шума**

Table 4. **Minimum and maximum noise level**

период	5					10					15				
число циклов	4	8	16	25	50	4	8	16	25	50	4	8	16	25	50
минимальный шум	0,10	0,10	0,13	0,16	0,22	0,13	0,14	0,19	0,21	0,21	0,15	0,21	0,20	0,24	0,24
максимальный шум	0,50	0,38	0,40	0,37	0,38	0,45	0,40	0,37	0,33	0,33	0,38	0,34	0,32	0,32	0,30
минимальный ожидаемый шум	0,10	0,13	0,13	0,18	0,24	0,13	0,18	0,22	0,23	0,24	0,19	0,22	0,22	0,26	0,27
максимальный ожидаемый шум	0,55	0,48	0,43	0,43	0,42	0,45	0,45	0,41	0,37	0,36	0,45	0,39	0,36	0,36	0,35

9. Влияние пропорции шумов разных типов на наблюдаемый уровень шума.

В ходе вычислительного эксперимента с двухуровневой моделью шума было замечено, что соотношение шумов разных типов по-разному влияет на уровень наблюдаемого шума. Некоторые результаты представлены в табл. 5, первые три столбца содержат значения уровня внесенного шума замены, вставки и удаления. В остальных столбцах указан диапазон относительных отклонений наблюдаемого общего уровня шума (вычисленного по расстоянию Левенштейна) от внесенного. Относительное отклонение вычислено по формуле

$$\omega = 100 - \frac{100 * \rho}{\alpha + \beta + \gamma},$$

где ρ – средний общий наблюдаемый уровень шума, α - уровень шума замены, β - вставки, γ - удаления.

Таблица 5. **Относительное отличие наблюдаемого шума от внесенного**

Table 5. **Relative difference between observed noise and introduced noise**

уровень шума замены	уровень шума вставки	уровень шума удаления	относительное отличие, период 5	относительное отличие, период 10	относительное отличие, период 25
0,01	0,01	0,01	7-14	9-35	7-8
0,01	0,01	0,10	0-2	0-5	5-9
0,05	0,01	0,10	3-9	2-8	1-2
0,10	0,01	0,01	3-17	1-8	2-6
0,10	0,05	0,05	8-17	14-22	14-16
0,10	0,10	0,10	25-30	27-30	27-30

Из таблицы видно, что на уровень общего наблюдаемого шума влияет как сумма уровней шумов всех типов, так и структура распределения уровней шума по типам шумов. Планируется в дальнейшем провести более детальный анализ влияния структуры шумов по типам на уровень как наблюдаемого общего шума, так и шумов каждого типа.

10. Заключение

В статье предложены две вероятностные модели шума, вносимого в циклическую символьную последовательность. В результате анализа данных, полученных в ходе вычислительного эксперимента, сделан вывод о том, что модель шума в каждом символе дает уровень шума, который может значительно отличаться от заданного. В отличие от этой модели, двухуровневая модель позволяет:

- обеспечить близкий к заданному уровень символьного шума;
- описывать единообразно шумы различных типов за счет использования вероятностной модели нижнего уровня, разыгрывание которой позволяет получить случайный символ алфавита кодирования;
- гибко описывать позиции шума в последовательности, это свойство обеспечивается тем, что случайная величина верхнего уровня выбирается самим исследователем и тем самым определяется частотная встречаемость подлежащих изменению позиций в исходной последовательности.

Предлагается использовать двухуровневую вероятностную модель для генерации синтетических почти периодических последовательностей с заданным уровнем шума в целях детального анализа свойств и чувствительности методов идентификации циклов в зашумленных символьных последовательностях.

Список использованных источников

[1] *Ma Sh., Hellerstein J. L.* Mining Partially Periodic Event Patterns with Unknown Periods // Proceedings 17th International Conference on Data Engineering. Heidelberg, Germany, 2001. Pp. 205-214. DOI: 10.1109/ICDE.2001.914829

[2] *He Z., Wang X.S., Lee B.S., Ling A. C. H.* Mining Partial Periodic Correlations in Time Series // Knowledge and Information Systems. 2008. Vol. 15, Issue 1. Pp. 31-54. DOI: 10.1007/s10115-006-0051-5

[3] *Elfeky M. G., Aref W. G., Elmagarmid A. K.* Periodicity Detection in Time Series Databases // IEEE Transactions on Knowledge and Data Engineering. 2005. Vol. 17, No. 7. Pp. 875-887. DOI: 10.1109/TKDE.2005.114

[4] *Kung-JiuanYang, Tzung-Pei Hong, Guo-Cheng Lan, Yuh-Min Chen.* A Two-Phase Approach for Mining Weighted Partial Periodic Patterns // Engineering Applications of Artificial Intelligence. 2014. Vol. 30. Pp. 225-234. DOI: 10.1016/j.engappai.2014.01.004

[5] *Ran He, Sen Yang, Jingyuan Yang, Jin Cao.* Automated Mining of Approximate Periodicity on Numeric Data: A Statistical Approach // Proceedings of the 2nd International Conference on Compute and Data Analysis (ICCDA 2018). ACM, New York, NY, USA, 2018. Pp. 20-27. DOI: 10.1145/3193077.3194509

[6] *Yuan H., Qian Y., Bai M.* Efficient Mining of Event Periodicity in Data Series / Li G., Yang J., Gama J., Natwichai J., Tong Y. (eds). Database Systems for Advanced Applications. DASFAA 2019. Lecture Notes in Computer Science. Vol. 11446. Springer, Cham, 2019. DOI: 10.1007/978-3-030-18576-3_8

[7] *Kiran R.U., Kitsuregawa M., Reddy P.K.* Efficient Discovery of Periodic-Frequent Patterns in Very Large Databases // Journal of Systems and Software. 2016. Vol. 112, Issue C. Pp.110-121. DOI: 10.1016/j.jss.2015.10.035

[8] *Venkatesh J. N., Kiran R. U., Reddy P. K., Kitsuregawa M.* Discovering Periodic-Frequent Patterns in Transactional Databases Using All-Confidence and Periodic-All-Confidence // Proceedings of the 27th International Conference on Database and Expert Systems Applications. Porto, Portugal, 2016. Part. I, LNCS 9827. Pp. 55-70. DOI: 10.1007/978-3-319-44403-1_4

[9] *Patel M., Modi N.* A Comprehensive Study on Periodicity Mining Algorithms // 2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC). Jalgaon, 2016. Pp. 567-575. DOI: 10.1109/ICGTSPICC.2016.7955365

[10] *Cao J., Drabeck L., He R.* Statistical Network Behavior Based Threat Detection // 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). Atlanta, GA, 2017. Pp. 420-425. DOI: 10.1109/INFCOMW.2017.8116413

[11] *Huang H., Liu F., Zha X., Xiong X., Ouyang T., Liu W.* et al. Robust Bad Data Detection Method for Microgrid Using Improved ELM and DBSCAN Algorithm // Journal of Energy Engineering. 2018. Vol. 144, Issue 3. Pp. 04018026. DOI:

10.1061/(ASCE)EY.1943-7897.0000544

[12] *Aydin B., Angryk R.* Spatiotemporal Frequent Pattern Mining on Solar Data: Current Algorithms and Future Directions // 2015 IEEE International Conference on Data Mining Workshop (ICDMW). Atlantic City, NJ, 2015. Pp. 575-581. DOI: 10.1109/ICDMW.2015.10

[13] *Dong S., Liu S., Zhao Y.* et al. An Innovative Model to Mine Asynchronous Periodic Pattern of Moving Objects // Multimedia Tools and Applications. 2019. Vol. 78, Issue 7. Pp. 8943-8964. DOI: 10.1007/s11042-018-6752-4

[14] *Li D., Yan W., Li W., Ren Z.* A Two-Tier Wind Power Time Series Model Considering Day-to-Day Weather Transition and Intraday Wind Power Fluctuations // IEEE Transactions on Power Systems. 2016. Vol. 31, No. 6. Pp. 4330-4339. DOI: 10.1109/TPWRS.2016.2531739

[15] *He Sh., Zhenxin Q.* Research on the Periodical Behavior Discovery of Funds in Anti-money Laundering Investigation // ICMLC '19 Proceedings of the 2019 11th International Conference on Machine Learning and Computing. 2019. Pp. 516-520. DOI: 10.1145/3318299.3318356

[16] *Гонсалес Р., Вудс Р.* Цифровая обработка изображений. Москва: Техносфера. 2006. 1072 с.

[17] *Уидроу Б., Стирнз С.* Адаптивная обработка сигналов. М.: Радиоисвязь. 1989. 440 с.

[18] *Rasheed F., Alhajj R.* STNR: A Suffix Tree Based Noise Resilient Algorithm for Periodicity Detection in Time Series Databases // Applied Intelligence. 2010. Vol. 32, Issue 3. Pp. 267-278. DOI: 10.1007/s10489-008-0144-9

[19] *Bjørnstad O. N., Viboud C.* Timing and Periodicity of Influenza Epidemics // Proceedings of the National Academy of Sciences. 2016. Vol. 113, No. 46. Pp. 12899-12901. DOI: 10.1073/pnas.1616052113

[20] *Chanda A. K.* et al. An Efficient Approach to Mine Flexible Periodic Patterns in Time Series Databases // Engineering Applications of Artificial Intelligence. 2015. Vol. 44. Pp. 46-63. DOI: 10.1016/j.engappai.2015.04.014

[21] *Korotkov E. V., Korotkova M. A.* Developing New Mathematical Method for Search of the Time Series Periodicity with Deletions and Insertions // Journal of Physics: Conference Series. 2017. Vol. 788, No. 1. Pp. 012019. DOI: 10.1088/1742-6596/788/1/012019.

[22] *Lin J.* et al. A Symbolic Representation of Time Series, with Implications for Streaming Algorithms // Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery. ACM, 2003. Pp. 2-11. DOI: 10.1145/882082.882086

[23] *Сметанин Ю. Г., Ульянов М. В.* Подход к определению характеристик колмогоровской сложности временных рядов на основе символьных описаний // Бизнес-информатика. 2013. № 2(24). С. 49-54. URL: https://elibrary.ru/item.asp?id=19526577 (дата обращения: 07.06.2019).

[24] *Левенштейн В. И.* Двоичные коды с исправлением выпадений, вставок и замещений символов // Доклады АН СССР. 1965. Т. 163, № 4. С. 845-848.

[25] *Гасфилд Д.* Строки, деревья и последовательности в алгоритмах: Информатика и вычислительная биология. СПб.: Невский диалект; БХВ–Петербург, 2003. 654 с.

Поступила 07.06.2019; принята к публикации 03.07.2019; опубликована онлайн 25.07.2019.

Об авторах:

Жукова Галина Николаевна, доцент Департамента программной инженерии, Факультет компьютерных наук, Национальный исследовательский университет «Высшая школа экономики» (101000, Россия, г. Москва, ул. Мясницкая, д. 20), кандидат физико-математических наук, доцент, ORCID: http://orcid.org/0000-0003-1835-7422, gzhukova@hse.ru

Сметанин Юрий Геннадиевич, главный научный сотрудник, Вычислительный центр им. А.А.Дородницына РАН, Федеральный исследовательский центр «Информатика и управление» Российской академии наук (119333, Россия, г. Москва, ул. Вавилова, д. 44, корп. 2), доктор физико-математических наук, ORCID: http://orcid.org/0000-0003-0242-6972, ysmetanin@rambler.ru

Ульянов Михаил Васильевич, ведущий научный сотрудник, Институт проблем управления им. В.А. Трапезникова Российской академии наук (117997, Россия, г. Москва, ул. Профсоюзная, д. 65), профессор кафедры алгоритмических языков, факультет вычислительной математики и кибернетики, Московский государственный университет имени М.В. Ломоносова (119991, Россия, г. Москва, ГСП-1, Ленинские горы, д. 1), доктор технических наук, профессор, ORCID: http://orcid.org/0000-0002-5784-9836, muljanov@mail.ru

Все авторы прочитали и одобрили окончательный вариант рукописи.

References

[1] Ma Sh., Hellerstein J.L. Mining Partially Periodic Event Patterns with Unknown Periods. *Proceedings 17th International Conference on Data Engineering.* Heidelberg, Germany. 2001; 205-214. (In Eng.) DOI: 10.1109/ICDE.2001.914829

[2] He Z., Wang X.S., Lee B.S., Ling A.C.H. Mining Partial Periodic Correlations in Time Series. *Knowledge and Information Systems.* 2008; 15(1):31-54. (In Eng.) DOI: 10.1007/s10115-006-0051-5

[3] Elfeky M.G., Aref W.G., Elmagarmid A.K. Periodicity Detection in Time Series Databases. *IEEE Transactions on Knowledge and Data Engineering.* 2005; 17(7):875-887. (In Eng.) DOI: 10.1109/TKDE.2005.114

[4] Kung-JiuanYang, Tzung-Pei Hong, Guo-Cheng Lan, Yuh-Min Chen. A Two-Phase Approach for Mining Weighted Partial Periodic Patterns. *Engineering Applications of Artificial Intelligence.* 2014; 30:225-234. (In Eng.) DOI: 10.1016/j.engappai.2014.01.004

[5] Ran He, Sen Yang, Jingyuan Yang, Jin Cao. Automated Mining of Approximate Periodicity on Numeric Data: A Statistical Approach. *Proceedings of the 2nd International Conference on Compute and Data Analysis (ICCDA 2018).* ACM, New York, NY, USA. 2018; 20-27. (In Eng.) DOI: 10.1145/3193077.3194509

[6] Yuan H., Qian Y., Bai M. Efficient Mining of Event Periodicity in Data Series. In: Li G., Yang J., Gama J., Natwichai J., Tong Y. (eds). *Database Systems for Advanced Applications. DASFAA 2019. Lecture Notes in Computer Science.* Vol. 11446. Springer, Cham, 2019. (In Eng.) DOI: 10.1007/978-3-030-18576-3_8

[7] Kiran R.U., Kitsuregawa M., Reddy P.K. Efficient Discovery

of Periodic-Frequent Patterns in Very Large Databases. *Journal of Systems and Software.* 2016; 112(C):110-121. (In Eng.) DOI: 10.1016/j.jss.2015.10.035

[8] Venkatesh J.N., Kiran R.U., Reddy P.K., Kitsuregawa M. Discovering Periodic-Frequent Patterns in Transactional Databases Using All-Confidence and Periodic-All-Confidence. *Proceedings of the 27th International Conference on Database and Expert Systems Applications.* Porto, Portugal, 2016. Part. I, LNCS 9827, pp. 55-70. (In Eng.) DOI: 10.1007/978-3-319-44403-1_4

[9] Patel M., Modi N. A Comprehensive Study on Periodicity Mining Algorithms. *2016 International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC).* Jalgaon. 2016; 567-575. (In Eng.) DOI: 10.1109/ICGTSPICC.2016.7955365

[10] Cao J., Drabeck L., He R. Statistical Network Behavior Based Threat Detection. *2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS).* Atlanta, GA. 2017; 420-425. (In Eng.) DOI: 10.1109/INFCOMW.2017.8116413

[11] Huang H., Liu F., Zha X., Xiong X., Ouyang T., Liu W. et al. Robust Bad Data Detection Method for Microgrid Using Improved ELM and DBSCAN Algorithm. *Journal of Energy Engineering.* 2018; 144(3):04018026. (In Eng.) DOI: 10.1061/(ASCE)EY.1943-7897.0000544

[12] Aydin B., Angryk R. Spatiotemporal Frequent Pattern Mining on Solar Data: Current Algorithms and Future Directions. *2015 IEEE International Conference on Data Mining Workshop (ICDMW).* Atlantic City, NJ. 2015; 575-581. (In Eng.) DOI: 10.1109/ICDMW.2015.10

[13] Dong S., Liu S., Zhao Y. et al. An Innovative Model to Mine Asynchronous Periodic Pattern of Moving Objects. *Multimedia Tools and Applications.* 2019; 78(7):8943-8964. (In Eng.) DOI: 10.1007/s11042-018-6752-4

[14] Li D., Yan W., Li W., Ren Z. A Two-Tier Wind Power Time Series Model Considering Day-to-Day Weather Transition and Intraday Wind Power Fluctuations. *IEEE Transactions on Power Systems.* 2016; 31(6):4330-4339. (In Eng.) DOI: 10.1109/TPWRS.2016.2531739

[15] He Sh., Zhenxin Q. Research on the Periodical Behavior Discovery of Funds in Anti-money Laundering Investigation. *ICMLC '19 Proceedings of the 2019 11th International Conference on Machine Learning and Computing.* 2019; 516-520. (In Eng.) DOI: 10.1145/3318299.3318356

[16] Gonzalez R.C., Woods R.E. Digital Image Processing. Addison-Wesley, Reading, MA., 1992. 716 pp. (In Eng.)

[17] Widrow B., Stearns S.D. Adaptive Signal Processing. Pearson, 1985. 496 pp. (In Eng.)

[18] Rasheed F., Alhajj R. STNR: A Suffix Tree Based Noise Resilient Algorithm for Periodicity Detection in Time Series Databases. *Applied Intelligence.* 2010; 32(3):267-278. (In Eng.) DOI: 10.1007/s10489-008-0144-9

[19] Bjørnstad O.N., Viboud C. Timing and Periodicity of Influenza Epidemics. *Proceedings of the National Academy of Sciences.* 2016; 113(46):12899-12901. (In Eng.) DOI: 10.1073/pnas.1616052113

[20] Chanda A.K. et al. An Efficient Approach to Mine Flexible Periodic Patterns in Time Series Databases. *Engineering Applications of Artificial Intelligence.* 2015; 44:46-63. (In Eng.) DOI: 10.1016/j.engappai.2015.04.014

[21] Korotkov E.V., Korotkova M.A. Developing New Mathematical Method for Search of the Time Series Periodicity with Deletions and Insertions. *Journal of Physics: Conference Series.* 2017; 788(1):012019. (In Eng.) DOI: 10.1088/1742-6596/788/1/012019.

[22] Lin J. et al. A Symbolic Representation of Time Series, with Implications for Streaming Algorithms. *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery.* ACM, 2003, pp. 2-11. (In Eng.) DOI: 10.1145/882082.882086

[23] Smetanin Yu.G., Ulyanov M.V. Determining the Characteristics of Kolmogorov Complexity of Time Series: An Approach Based on Symbolic Descriptions. *Business Informatics.* 2013; 2(24):49-54. Available at: https://elibrary.ru/item.asp?id=19526577 (accessed 07.06.2019). (In Russ., abstract in Eng.)

[24] Levenshtein V.I. Binary Codes Capable of Correcting Deletions, Insertions, and Reversals. *Soviet Physics Doklady.* 1966; 10(8):707-710. Available at: https://nymity.ch/sybilhunting/pdf/Levenshtein1966a.pdf (accessed 07.06.2019). (In Eng.)

[25] Gusfield D. Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology. Cambridge University Press, 1997. 556 pp. (In Eng.)

Submitted 07.06.2019; revised 03.07.2019;
published online 25.07.2019.

About the authors:

Galina N. Zhukova, Associate Professor of the School of Software Engineering, Faculty of Computer Science, National Research University Higher School of Economics (20 Myasnitskaya St., Moscow 101000, Russia), Ph.D. (Phys.-Math.), Associate Professor, ORCID: http://orcid.org/0000-0003-1835-7422, gzhukova@hse.ru

Yuri G. Smetanin, Chief researcher of the Dorodnicyn Computing Centre of RAS, Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences (44-2 Vavilov St., Moscow 119333, Russia), Dr. Sci. (Phys.-Math.), ORCID: http://orcid.org/0000-0003-0242-6972, ysmetanin@rambler.ru

Mikhail V. Uljanov, Leading scientist researcher, V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences (65 Profsoyuznaya St., Moscow 117997, Russia), Professor of the Department of Algorithmic Languages, Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University (1, Leninskie gory, Moscow 119991, Russia), Dr.Sci. (Engineering), Professor, ORCID: http://orcid.org/0000-0002-5784-9836, muljanov@mail.ru

All authors have read and approved the final manuscript.