УДК 004.272.25, 004.421, 004.032.24 DOI: 10.14529/cmse230202

ПОИСК АНОМАЛИЙ В СЕНСОРНЫХ ДАННЫХ ЦИФРОВОЙ ИНДУСТРИИ С ПОМОЩЬЮ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛЕНИЙ*

© 2023 Я.А. Краева

Южно-Уральский государственный университет
(454080 Челябинск, пр. им. В.И. Ленина, д. 76)
E-mail: kraevaya@susu.ru

Поступила в редакцию: 20.09.2022

В статье представлены результаты исследований по поиску аномалий в сенсорных данных из различных приложений цифровой индустрии. Рассматриваются временные ряды, полученные при эксплуатации деталей машин, показания датчиков, установленных на металлургическом оборудовании, и показания температурных датчиков в системе умного управления отоплением зданий. Аномалии, найденные в таких данных, свидетельствуют о нештатной ситуации, отказах, сбоях и износе технологического оборудования. Аномалия формализуется как диапазонный диссонанс — подпоследовательность временного ряда, расстояние от которой до ее ближайшего соседа не менее наперед заданного аналитиком порога. Ближайшим соседом данной подпоследовательности является такая подпоследовательность ряда, которая не пересекается с данной и имеет минимальное расстояние до нее. Поиск диссонансов выполняется с помощью параллельного алгоритма для графического процессора, ранее разработанного автором данной статьи. Для визуализации найденных аномалий предложены метод построения тепловой карты диссонансов, имеющих различные длины, и алгоритм нахождения в построенной тепловой карте наиболее значимых диссонансов независимо от их длин.

Ключевые слова: временной ряд, сенсорные данные, поиск аномалий, диссонанс, параллельный алгоритм, графический процессор, CUDA.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Краева Я.А. Поиск аномалий в сенсорных данных цифровой индустрии с помощью параллельных вычислений // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2023. Т. 12, № 2. С. 47–61. DOI: 10.14529/cmse230202.

Введение

В настоящее время поиск аномалий временных рядов является одной из актуальных задач в широком спектре предметных областей, связанных с обработкой сенсорных данных [1]. В приложениях цифровой индустрии [2] и Интернета вещей [3, 4] датчики киберфизических систем имеют высокую дискретность снятия показаний (десятки–сотни раз в секунду) и за короткое время продуцируют временные ряды, состоящие из сотен миллионов элементов. Аномалии, найденные в данных сенсоров, свидетельствуют о нештатной ситуации, отказах, сбоях и износе технологического оборудования. Обнаружение аномалий может использоваться для заблаговременного уведомления оператора технологического процесса и организации предиктивного технического обслуживания и ремонта оборудования, что в конечном итоге увеличивает остаточный ресурс этого оборудования.

В данном исследовании поиск аномалий предполагает нахождение подпоследовательностей временного ряда, которые наименее похожи на все остальные подпоследовательности ряда. Одним из наиболее эффективных подходов к поиску аномалий является концепция

*Статья рекомендована к публикации программным комитетом Международной научной конференции «Параллельные вычислительные технологии (ПаВТ) 2023».

диапазонного диссонанса (range discord) [5, 6]. Диапазонный диссонанс (далее для краткости — диссонанс) представляет собой подпоследовательность ряда, расстояние от которой до ее ближайшего соседа не менее заданного порога, являющегося параметром алгоритма. Ближайшим соседом данной подпоследовательности является такая подпоследовательность ряда, которая не пересекается с данной и имеет минимальное расстояние до нее. Применение концепции диссонансов для поиска аномалий требует от аналитика интуитивно понятных параметров (длина подпоследовательности и порог расстояния), в отличие от альтернативных подходов, требующих более трех не всегда интуитивно понятных параметров [7].

Авторы концепции диссонанса предложили последовательный алгоритм DRAG (Discord Range Aware Gathering) [6] поиска диссонансов временного ряда. Ранее автором настоящей статьи был разработан алгоритм PD3 (Parallel DRAG-based Discord Discovery) [8], представляющий собой параллельную версию алгоритма DRAG. В данной статье показано, как алгоритм PD3 может быть применен для поиска аномалий в приложениях цифровой индустрии. Рассматриваются сенсорные данные, полученные при эксплуатации деталей машин, показания датчиков, установленных на металлургическом оборудовании, и показания температурных датчиков в системе умного управления отоплением зданий. Для визуализации найденных аномалий предлагается метод построения тепловой карты диссонансов, имеющих различные длины, и предложен способ нахождения в построенной тепловой карте наиболее значимых диссонансов независимо от их длин.

Статья организована следующим образом. В разделе 1 приводятся формальные определения и краткое описание параллельного алгоритма PD3. В разделе 2 представлены метод построения тепловой карты диссонансов и алгоритм нахождения наиболее значимых диссонансов. В разделе 3 описаны результатов исследований по поиску аномалий в сенсорных данных цифровой индустрии. Заключение подводит итоги исследования.

1. Параллельный алгоритм поиска диссонансов PD3

1.1. Формальные определения и обозначения

В данном разделе приводятся обозначения и определения используемых терминов в соответствии с работами [5, 6].

Временной ряд (time series) T представляет собой последовательность хронологически упорядоченных вещественных значений:

$$T = \{t_i\}_{i=1}^{n}, \quad t_i \in \mathbb{R}. \tag{1}$$

Число n обозначается $|T|$ и называется длиной ряда.

Подпоследовательность (subsequence) $T_{i,\,m}$ временного ряда T представляет собой непрерывный промежуток из m элементов, начиная с позиции i:

$$T_{i,m} = \{t_k\}_{k=i}^{i+m-1}, \quad 1 \leqslant m \ll n, \quad 1 \leqslant i \leqslant n - m + 1. \tag{2}$$

Множество всех подпоследовательностей ряда T, имеющих длину m, обозначим как S_T^m, а мощность такого множества за N, $N = |S_T^m| = n - m + 1$.

Подпоследовательности $T_{i,\,m}$ и $T_{j,\,m}$ ряда T называются *непересекающимися (non-self match)*, если $|i - j| \geqslant m$. Подпоследовательность, которая является непересекающейся к данной подпоследовательности C, будем обозначать как M_C.

Подпоследовательность D ряда T является *диапазонным диссонансом (range discord)*, если

$$\forall M_D \in T \ \min\big(\mathrm{Dist}(D, M_D)\big) > r, \tag{3}$$

где $\mathrm{Dist}(\cdot\,,\cdot\,)$ представляет собой неотрицательную симметричную функцию, порог расстояния r — наперед заданный параметр. Другими словами, некая подпоследовательность ряда является диапазонным диссонансом, если ее ближайший сосед (ближайшая и не пересекающаяся с ней подпоследовательность) находится на расстоянии не менее чем r. Далее для краткости будем использовать термин «диссонанс», подразумевая диапазонный диссонанс, если не указано обратное.

Алгоритм PD3 предполагает, что обрабатываемые подпоследовательности временного ряда предварительно подвергнуты z-нормализации. *Z-нормализация* подпоследовательности (ряда) T представляет собой подпоследовательность (ряд) $\hat{T} = (\hat{t}_1, \ldots, \hat{t}_m)$, элементы которого вычисляются следующим образом:

$$\hat{t}_i = \frac{t_i - \mu}{\sigma}, \quad \mu = \frac{1}{m}\sum_{i=1}^{m} t_i, \quad \sigma = \sqrt{\frac{1}{m}\sum_{i=1}^{m} t_i^2 - \mu^2}. \tag{4}$$

В данном исследовании в качестве функции $\mathrm{Dist}(\cdot\,,\cdot\,)$ используется квадрат нормированного евклидова расстояния, обозначаемого как $\mathrm{ED}^2_{\mathrm{norm}}(\cdot\,,\cdot\,)$. Вычисление указанного расстояния с помощью формулы, предложенной в работе [9], выполняется быстрее, чем с использованием формулы (4):

$$\mathrm{ED}^2_{\mathrm{norm}}(T_{i,m}, T_{j,m}) = \mathrm{ED}^2(\hat{T}_{i,m}, \hat{T}_{j,m}) = 2m\left(1 - \frac{< T_{i,m}, T_{j,m} > - m\mu_i\mu_j}{m\sigma_i\sigma_j}\right), \tag{5}$$

где подпоследовательности $T_{i,m}$ и $T_{j,m}$ рассматриваются как вектора в евклидовом пространстве $\mathbb{R}^m$, μ_i и μ_j, σ_i и σ_j — среднее арифметическое и стандартное отклонение указанных векторов соответственно.

1.2. Принципы реализации алгоритма

Алгоритм PD3 распараллеливает вычислительную схему алгоритма DRAG [6], которая заключается в следующем. DRAG сначала выполняет отбор кандидатов в диссонансы, а затем очищает полученное множество от ложноположительных кандидатов. На этапе отбора множество кандидатов $\mathcal{C}$ инициализируется первой подпоследовательностью ряда T. Далее алгоритм сканирует ряд с помощью скользящего окна длины m и для каждой подпоследовательности $s \in S_T^m$ выполняет проверку, что каждый кандидат $c \in \mathcal{C}$ является диссонансом. Кандидат c, не прошедший проверку, удаляется из $\mathcal{C}$. По завершении проверки s либо добавляется в множество кандидатов, либо удаляется из него. На этапе очистки сначала для каждого кандидата из $\mathcal{C}$ расстояние до его ближайшего соседа полагается $+\infty$. Затем алгоритм сканирует ряд с помощью скользящего окна длины m, вычисляя расстояние между каждой подпоследовательностью $s \in S_T^m$ и каждым кандидатом c. Если расстояние меньше r, то кандидат является удаляется из $\mathcal{C}$ как ложноположительный. Если указанное расстояние меньше текущего минимального расстояния до ближайшего соседа, то текущий минимум расстояния до ближайшего соседа обновляется этим значением.

Алгоритм PD3 [8], используя параллелизм по данным, распараллеливает на графическом процессоре этапы отбора и очистки алгоритма DRAG. Временной ряд сегментируется,

и отдельный блок нитей GPU обрабатывает свой сегмент. На этапе отбора схема работы алгоритма PD3 выглядит следующим образом. Блок нитей полагает все подпоследовательности своего сегмента кандидатами и обрабатывает те из них, которые расположены справа от него и не пересекаются с кандидатами. Если расстояние от кандидата до подпоследовательности меньше r, то они заведомо не являются диссонансами и отбрасываются. Если все кандидаты отброшены, блок завершает работу. Блок нитей обрабатывает подпоследовательности ряда порциями, количество элементов в которых равно длине сегмента. При этом первая порция таких элементов начинается с m-го элемента в сегменте, что позволяет избежать избыточных проверок на пересечение кандидатов и подпоследовательностей обрабатываемой порции. Обработка порции заключается в вычислении расстояний от всех подпоследовательностей сегмента, назначенного блоку, до всех подпоследовательностей данной порции, на основе формулы (5), и выполняется следующим образом. Сперва нити блока вычисляют скалярные произведения между первой подпоследовательностью сегмента и всеми подпоследовательностями текущей порции, сохраняя результат в массиве в разделяемой памяти GPU. Далее вычисляются расстояния между первой подпоследовательностью порции и всеми подпоследовательностями сегмента, при этом используются полученные ранее результаты. Вычисленное расстояние используется для отбрасывания ложноположительных кандидатов в сегменте и текущей порции. Очистка найденных кандидатов выполняется в алгоритме PD3 аналогично описанной выше процедуре отбора. В очистке задействуются сегменты ряда с непустыми множествами кандидатов, при этом обрабатываются подпоследовательности ряда, не пересекающиеся с кандидатами и находящиеся слева от сегмента. Если расстояние от кандидата до подпоследовательности меньше r, то кандидат отбрасывается.

2. Визуализация диссонансов

Визуализация является важной и неотъемлемой частью решения задач интеллектуального анализа данных, поскольку обеспечивает аналитику наглядное представлением исследуемых данных и результатов анализа, являющееся основой выявления скрытых закономерностей для принятия стратегически важных решений [10].

В данном разделе представлен новый метод визуализации диссонансов временного ряда, который предполагает построение тепловой карты, где степень аномальности диссонансов отражается посредством интенсивности цвета. Далее в разделах 2.1 и 2.2 описаны детали построения тепловой карты и способ нахождения наиболее значимых аномалий на основе построенной карты соответственно.

2.1. Построение тепловой карты диссонансов

Пусть имеется временной ряд T длины n, в котором с помощью алгоритма PD3 осуществляется поиск диссонансов длины m, принимающей значения в заданном диапазоне $minL \leqslant m \leqslant maxL$ ($minL \leqslant maxL \ll n$). Последовательно запуская алгоритм PD3 $maxL - minL + 1$ раз для каждого значения длины из указанного диапазона, получим итоговое множество диссонансов $\mathcal{D} = \bigcup\limits_{m=minL}^{maxL} D_m$, где D_m — подмножество диссонансов, имеющих длину m.

Далее рассмотрим матричный профиль временного ряда T для длины подпоследовательности m. *Матричный профиль (matrix profile)* [11] временного ряда T для длины подпоследовательности m представляет собой временной ряд $MP_m \in \mathbb{R}^{n-m+1}$, элементами кото-

рого являются расстояния от соответствующей подпоследовательности ряда до ближайшего не пересекающегося с ней соседа:

$$MP_m(i) = \mathrm{Dist}(T_{i,\,m},\, Neighbor), \quad Neighbor = \arg \min_{T_{j,\,m} \in S_T^m} \mathrm{Dist}(T_{i,\,m},\, T_{j,\,m}), \quad |i-j| \geqslant m. \quad (6)$$

Затем возьмем матричные профили ряда T для всех длин подпоследовательности из диапазона $minL..maxL$ в порядке возрастания длин и построим из них матрицу профилей $MMP \in \mathbb{R}^{(maxL-minL+1)\times(n-minL)}$. В качестве строки указанной матрицы фигурирует отдельный матричный профиль ряда, где обнулены элементы, соответствующие подпоследовательностям ряда, которые не являются диссонансами (индексы строки и столбца матрицы показывают соответственно длину диссонанса и его индекс в ряде):

$$MMP(m,\, i) = \begin{cases} MP_{minL+m-1}(i), & T_{i,\,minL+m-1} \in D_{minL+m-1} \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

Отметим, что матрица MMP является естественным следствием запусков алгоритма PD3 и не требует отдельных вычислений матричных профилей ряда для рассматриваемых длин подпоследовательности (например, с помощью алгоритма [12]). Далее определим *тепловую карту диссонансов (discord heatmap)* как матрицу $heatmap \in \mathbb{R}^{(maxL-minL+1)\times(n-minL)}$, получаемую нормированием элементов в каждой строке матрицы $MMP(m, \cdot)$ с помощью множителя $\frac{1}{2m}$:

$$heatmap(m,\, i) = \frac{MMP(m,\, i)}{2m}. \quad (8)$$

Указанный нормирующий множитель обеспечивает приведение значений элементов тепловой карты к диапазону от 0 до 1, поскольку доказано [9], что положительная корреляция по Пирсону двух векторов $x,\, y \in \mathbb{R}^m$ и нормализованное евклидово расстояние между ними связаны следующим соотношением:

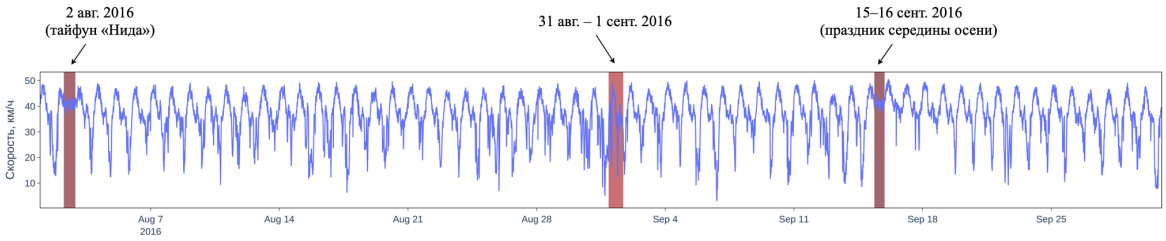
$$PearsonCorr(x, y) = 1 - \frac{\mathrm{ED}^2_{\mathrm{norm}}(x,\, y)}{2m}. \quad (9)$$

Таким образом, каждый элемент $heatmap(m,\, i)$ представляет собой оценку аномальности диссонанса $T_{i,\,m} \in D_m$, нормированную по всем диссонансам множества $\mathcal{D}$. В тепловой карте диссонансов используется один цвет (например, красный), и интенсивность цвета в пикселе (m, i) прямо пропорциональна оценке диссонанса $T_{i,\,m} \in D_m$. При этом нормирование обеспечивает на одной тепловой карте корректную визуализацию оценок аномальности диссонансов различной длины.

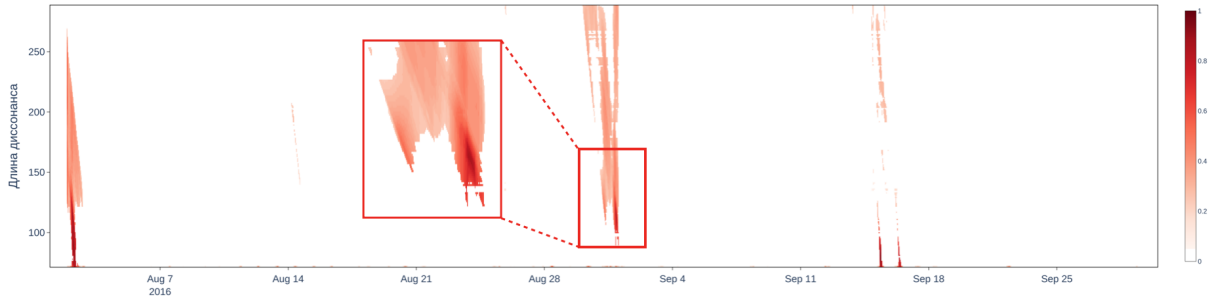
2.2. Ранжирование диссонансов различной длины

Тепловая карта диссонансов, описанная выше, представляет собой удобный для аналитика инструмент визуализации диссонансов заданного временного ряда, имеющих различную длину. Тем не менее, для аналитика также важно ранжирование найденных диссонансов по их практической значимости вне зависимости от их длины. Далее представлен алгоритм нахождения top-k наиболее значимых диссонансов, который основан на простой идее отдавать предпочтение диссонансам, которые имеют бóльшую оценку, исключая при этом пересекающиеся диссонансы (см. Алг. 1).

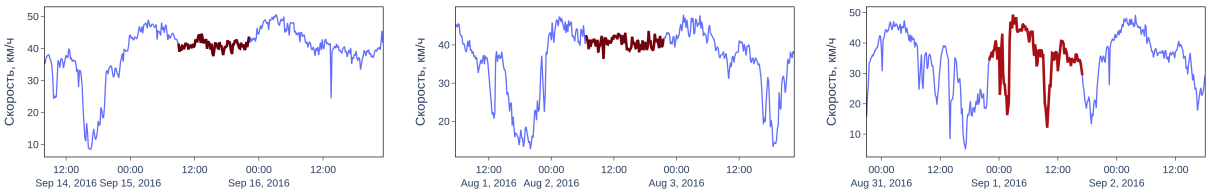
Алгоритм получает на входе тепловую карту диссонансов $heatmap \in \mathbb{R}^{(maxL-minL+1)\times(n-minL)}$ и число K искомых диссонансов, а на выходе формирует



(a) Временной ряд



(b) Тепловая карта диссонансов



Top-1 диссонанс, $m = 78$ Top-2 диссонанс, $m = 87$ Top-3 диссонанс, $m = 116$

(c) Найденные top-3 интересных диссонанса различных длин

Рис. 1. Результаты визуализации и подхода нахождения наиболее интересных диссонансов

Алг. 1 TopInterestDiscords (in: $heatmap$, K; out: $InterestD$)

$\overline{Scores} \leftarrow \{score_i\}_{i=1}^{n-minL+1}$, где $score_i = \max\limits_{minL \leqslant m \leqslant maxL} heatmap(m, i)$

$\overline{Lengths} \leftarrow \{length_i\}_{i=1}^{n-minL+1}$, где $length_i = \arg\max\limits_{minL \leqslant m \leqslant maxL} heatmap(m, i)$

$\overline{Indexes} \leftarrow \{i\}_{i=1}^{n-minL+1}$

Sort($\overline{Scores}, \overline{Lengths}, \overline{Indexes}$) ▷ Сортировка векторов по убыванию значений в $\overline{Scores}$

$InterestD \leftarrow \{(T_{\overline{Indexes}(1), \overline{Lengths}(1)}, \overline{Scores}(1))\}$; $k \leftarrow 2$

while ($|InterestD| < K$) **or** ($k < N$) **do**

  $score \leftarrow \overline{Scores}(k)$; $length \leftarrow \overline{Lengths}(k)$; $index \leftarrow \overline{Indexes}(k)$

  for each $T_{i,m} \in InterestD$ **do**

    if $\min(length, m) < |i - index| < \max(length, m)$ **then** ▷ Проверка пересечения

      $InterestD \leftarrow InterestD \cup \{(T_{index,\, length}, score)\}$

    else

      break

  $k \leftarrow k + 1$

return $InterestD$

множество $Interest\mathcal{D}$, элементами которого являются пары вида $(T_{i,m}, score)$, где диссонанс $T_{i,m} \in D_m$, $score$ — оценка диссонанса. Алгоритм выполняется следующим образом. Сначала формируются вектора $Scores, Lengths \in \mathbb{R}^{n-minL+1}$, в которых i-й элемент хранит соответственно максимальную оценку диссонанса с индексом i и длину такого диссонанса. Вектор $Indexes \in \mathbb{R}^{n-minL+1}$ хранит индексы диссонансов. Далее вектор $Scores$ сортируется по убыванию, а вектора $Lengths$ и $Indexes$ упорядочиваются на основе результата такой сортировки. В итоге указанные три вектора совместно представляют сведения о диссонансах множества $\mathcal{D}$ в порядке убывания оценок входящих в него диссонансов (строки 1–4 в Алг. 1). Искомое множество диссонансов инициализируется диссонансом с максимальной оценкой (строка 5 в Алг. 1). Затем выполняется просмотр диссонансов множества $\mathcal{D}$ в полученном порядке (цикл в строках 6–13 в Алг. 1). Если рассматриваемый диссонанс не пересекается ни с одним элементом из $Interest\mathcal{D}$, то он добавляется в него. Сканирование продолжается до тех пор, пока не найдено K наиболее значимых диссонансов или исчерпано множество $\mathcal{D}$.

Проиллюстрируем работу описанного алгоритма на следующем реальном примере. На рис. 1a представлен пример временного ряда [13], который содержит данные о скорости городского трафика на одном из участков городской автомагистрали Гуанчжоу (Китай), измерявшиеся каждые 10 мин. с 1 августа по 30 сентября 2016 г. В ряде выделены красным цветом диссонансы различной длины, которые были найдены с помощью алгоритма PD3. На рис. 1b показана тепловая карта найденных диссонансов. Далее, на рис. 1c показаны три наиболее важных диссонанса (имеющие различную длину), отобранные с помощью описанного выше алгоритма TopInterestDiscords. Можно видеть, что у двух из указанных диссонансов время возникновения совпадает с нетипичными событиями: тайфун Нида и отмечаемый в Китае Праздник середины осени.

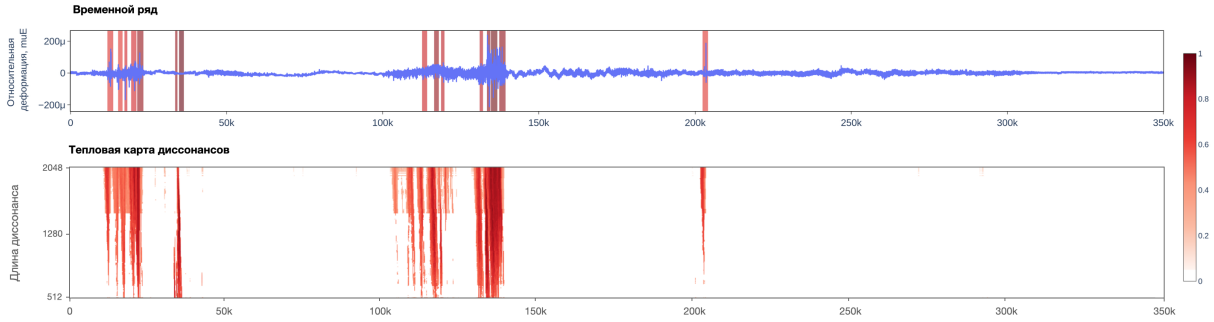
3. Поиск диссонансов в сенсорных данных цифровой индустрии

В данном разделе приведены результаты исследований по применению разработанного автором данной статьи алгоритма для поиска диссонансов в сенсорных данных из различных областей цифровой индустрии. Указанные исследования проведены на оборудовании Лаборатории суперкомпьютерного моделирования ЮУрГУ [14]: графический процессор NVIDIA Tesla V100 SXM2 (5 120 ядер с тактовой частотой 1.3 GHz, пиковая производительность 7 TFLOPS для чисел с двойной точностью).

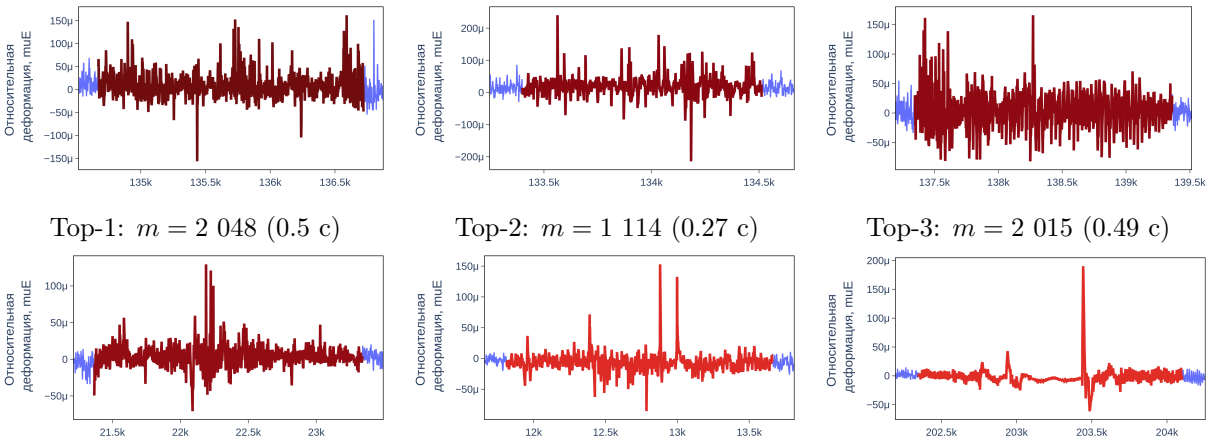
3.1. Деформации в механизме сцепки вагонов трамвая

Первое исследование связано с поиском диссонансов в сенсорных данных, полученных с тензометрического датчика, установленного на механизме сцепки вагонов трамвая. Данные указанного датчика представляют собой значения относительной деформации, связанные со значениями напряжений, возникающими в процессе эксплуатации механизма сцепки вагонов. Использованные в исследовании данные сняты в течение одной рабочей смены трамвая, следующего по одному маршруту, с частотой дискретизации 4 096 Гц в течение 1.5 мин. (350 000 точек).

На рис. 2a представлены временной ряд и тепловая карта найденных с помощью разработанного автором данной статьи параллельного алгоритма диссонансов, имеющих длину в диапазоне 512..2 048. На рис. 2b представлены примеры шести наиболее значимых дис-



(a) Временной ряд и тепловая карта диссонансов



Top-1: $m = 2\,048$ (0.5 с) Top-2: $m = 1\,114$ (0.27 с) Top-3: $m = 2\,015$ (0.49 с)

Top-4: $m = 1\,975$ (0.48 с) Top-5: $m = 1\,852$ (0.45 с) Top-6: $m = 1\,756$ (0.43 с)

(b) Примеры найденных top-k диссонансов различных длин

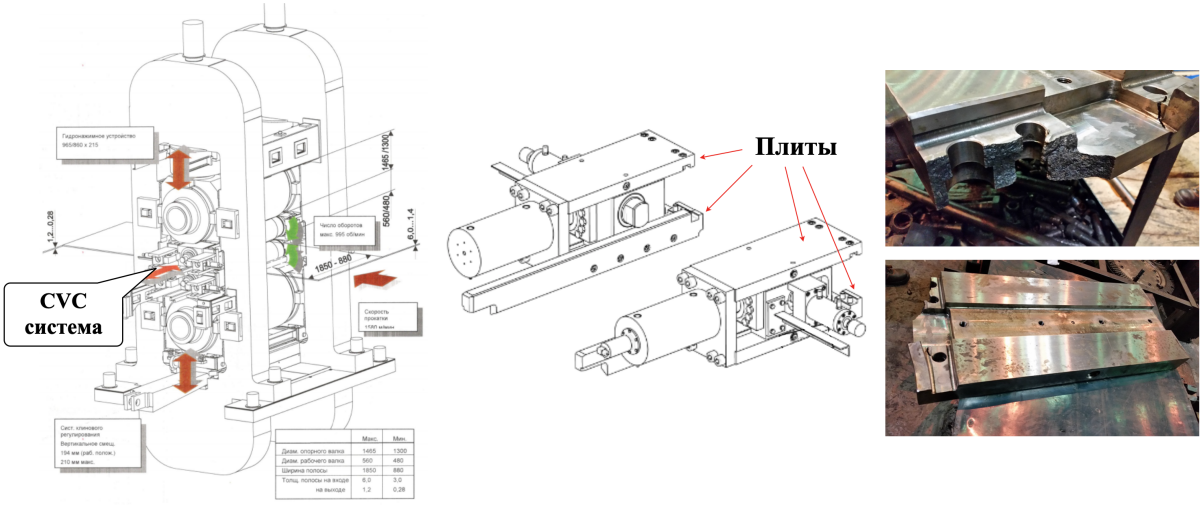
Рис. 2. Обнаружение аномалий во временном ряде относительных деформаций механизма стыковки вагонов трамвая

сонансов. Найденные диссонансы позволяют видеть, что при эксплуатации узла сцепки вагонов имеет место работа в экстремальных условиях с высокими амплитудами отклонения знакопеременной нагрузки. Очевидно, что при длительной работе в подобных условиях остаточный ресурс узла сцепки вагонов сокращается.

3.2. Разрушение плит системы профилировки валков стана холодной прокатки

Второе исследование связано с поиском диссонансов во временных рядах, полученных с сенсоров, установленных на стане холодной прокатки металлургического завода. Холодная прокатка сопровождается воздействием больших усилий и напряжений на основные рабочие и вспомогательные элементы стана, поэтому они быстро изнашиваются и часто ломаются, что негативно влияет на качество выпускаемой продукции. Для повышения качества металлопродукции и совершенствования технологии производства холоднокатанных полос каждая клеть стана оснащается системой регулировки профиля полос посредством осевой сдвижки выпукло-вогнутых рабочих валков CVC (Continuously Variable Curvature) [15] (см. рис. 3а). Система CVC позволяет добиться уменьшения разнотолщинности по профилю, по краям и по центру холоднокатанных полос и высокой плоскостности по всей ширине листа. Однако при эксплуатации системы CVC имеют место частые поломки плит, по которым происходит движение системы CVC совместно с рабочими валками клети в горизонтальном

направлении (см. рис. 3b). Причиной подобных поломок, предположительно, являются высокие значения знакопеременных нагрузок изгибающих моментов, вызванных изменениями напряженности клети. Указанные нагрузки приводят к появлению в наиболее нагруженных частях плит CVC трещин, которые ведут к разрушению плит (см. рис. 3c).



(a) Клеть стана холодной прокатки

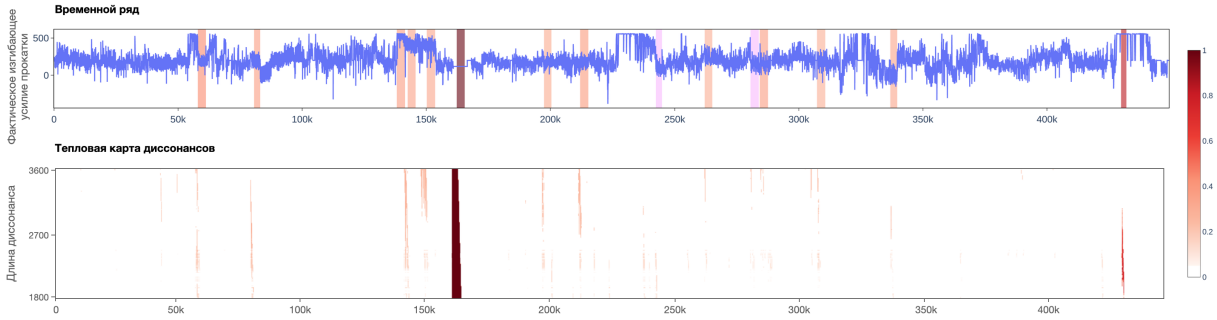
(b) CVC система рабочего валка клети стана

(c) Разрушение плит CVC системы

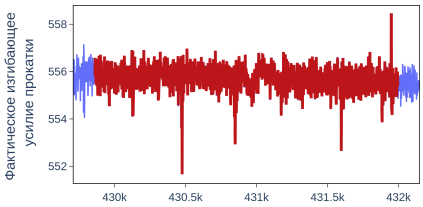
Рис. 3. Проблема поломки плит CVC системы стана холодной прокатки

В исследовании использовались данные сенсоров, установленные на одной из клетей стана холодной прокатки, которые измеряли различные показатели с частотой 1 раз в секунду в течение 6 месяцев, а также сведения о простоях клетей стана вследствие технического обслуживания или ремонта плит CVC за указанный период. В силу ранее сделанного предположения о высокой напряженности клети как о наиболее вероятной причине поломок для исследования был выбран временной ряд показаний датчика, измеряющего фактическое изгибающее усилие прокатки, в котором исключены периоды простоев стана, не связанных с поломками плит. Полученный ряд представлен на рис. 4a. В данном временном ряде выполнялся поиск диссонансов различных длин из диапазона от 30 минут до 1 часа (т.е. $minL = 1\ 800$ и $maxL = 3\ 600$).

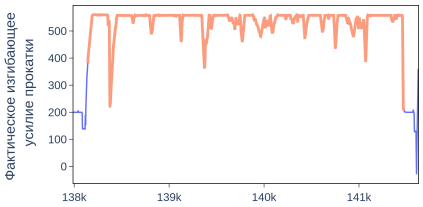
На рис. 4a представлены результаты работы алгоритма PD3 по поиску top-15 диссонансов, которые выделены красным цветом. Далее из них были отобраны 4 следующих диссонанса, которые наиболее точно отражают причины возникновения критических ситуаций при работе плит CVC системы (см. рис. 4b). Первый диссонанс показывает, что на плиту длительное время действуют напряжения, намного превышающие допустимые значения, вызванные напряженностью клети стана в отсутствии прокатываемого металла. Остальные три диссонанса соответствуют напряженному состоянию в плитах при знакопеременных нагрузках и критических значениях давлений на клеть при обработке проката. Отобранные диссонансы показывают, что в межремонтные периоды плиты, подвергаемые экстремальным знакопеременным нагрузкам, с большой вероятностью могут получить повреждения в виде трещин, приводящих к поломками и выходу плит из строя. Найденные диссонансы могут в дальнейшем быть использованы как паттерны снижения остаточного ресурса плит CVC системы и предсказания их поломок.



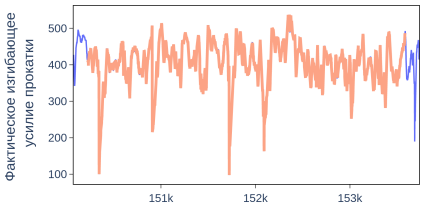
(a) Временной ряд фактического изгибающего усилия прокатки



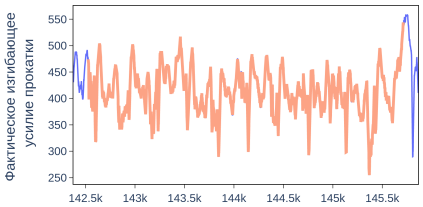
Top-3: $m = 2\ 898$ (48 мин. 18 с)



Top-5: $m = 3\ 345$ (55 мин. 45 с)



Top-7: $m = 3\ 576$ (59 мин. 36 с)



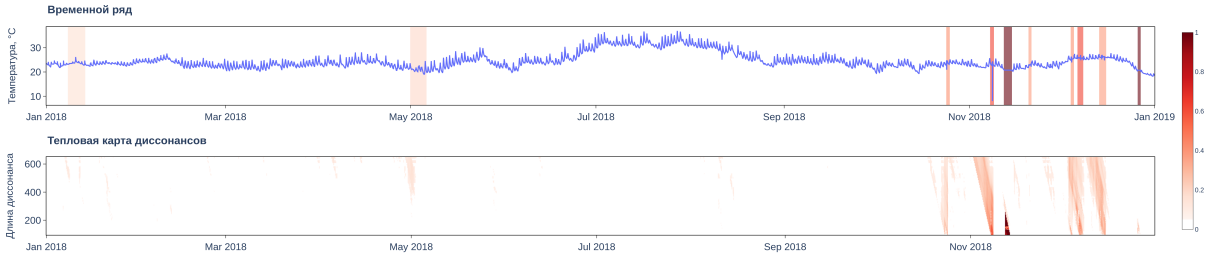
Top-10: $m = 3\ 187$ (53 мин. 6 с)

(b) Примеры найденных top-k диссонансов различных длин

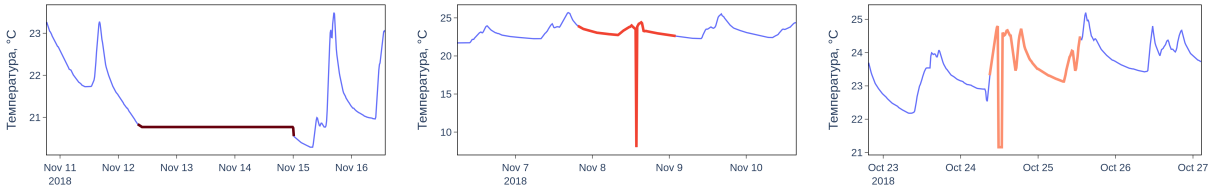
Рис. 4. Обнаружение аномалий в показаниях датчиков, установленных на металлургическом оборудовании

3.3. Нештатные ситуации в системе умного управления отоплением зданий

Третье исследование направлено на поиск диссонансов в показаниях температурных датчиков, которые являются частью интеллектуальной системы ПолиТЭР [16] управления теплоснабжением, установленной в Южно-Уральском государственном университете (ЮУрГУ). ПолиТЭР выполняет мониторинг инженерных систем зданий кампуса ЮУрГУ и управляет режимами их работы на основе анализа показаний проводных и беспроводных датчики. Для исследования были взяты показания беспроводного датчика, установленного в одной из учебных аудиторий ЮУрГУ (см. рис. 5a), за 2018 год при частоте снятия показаний 1 раз в 15 мин. В данном временном ряде выполнялся поиск диссонансов различных длин из диапазона от 1 дня до 1 недели (т.е. $minL = 96$ и $maxL = 672$). Примеры некоторых аномалий из списка top-10, полученного с помощью алгоритм PD3, представлены на рис. 5b. Первая аномалия показывает, что датчик, скорее всего, вышел из строя, поскольку в течение определенного времени передавалась постоянная температура. В остальных случаях аномалии свидетельствуют о влиянии человеческого фактора на энергоэффективность системы теплоснабжения зданий университета: вероятно, резкое изменение температуры вызвано сознательно либо случайно открытыми окнами в помещении.



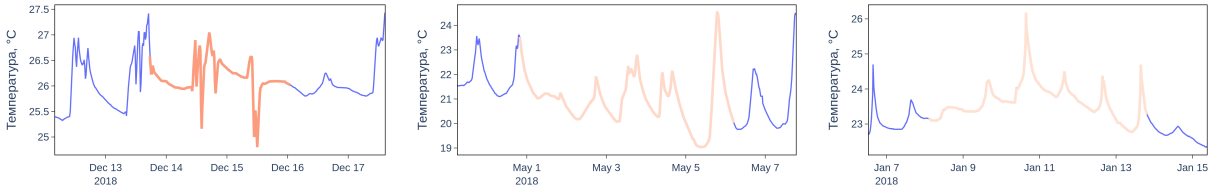
(a) Временной ряд и тепловая карта диссонансов



Top-1: $m = 258$
(2 дня 16 час. 30 мин.)

Top-3: $m = 122$
(1 день 6 час. 30 мин.)

Top-6: $m = 113$
(1 день 4 час. 15 мин.)



Top-7: $m = 219$
(2 дня 6 час. 45 мин.)

Top-9: $m = 504$
(5 дней 6 час.)

Top-10: $m = 650$
(6 дней 18 час. 30 мин.)

(b) Примеры найденных top-k диссонансов различных длин

Рис. 5. Обнаружение аномалий в показаниях температурных датчиков в системе умного управления отоплением зданий

Заключение

В статье рассмотрена проблема поиска аномалий в сенсорных данных, которая в настоящее время возникает в широком спектре предметных областей: цифровая индустрия, Интернет вещей и др. Аномалии, найденные в данных сенсоров, свидетельствуют о нештатной ситуации, отказах, сбоях и износе технологического оборудования. Оперативное обнаружение аномалий может использоваться для заблаговременного уведомления оператора технологического процесса и организации профилактического обслуживания и ремонта оборудования. В данном исследовании аномалия формализуется как диапазонный диссонанс [5, 6] — подпоследовательность ряда, расстояние от которой до ее ближайшего соседа не менее наперед заданного аналитиком порога. Ближайшим соседом данной подпоследовательности является такая подпоследовательность ряда, которая не пересекается с данной и имеет минимальное расстояние до нее.

В статье представлены результаты исследований по поиску диссонансов во временных рядах, которые представляют собой показания сенсоров из реальных предметных областей: деформации механизма стыковки вагонов трамвая, деформации плиты системы регулировки профиля полос стана холодной прокатки металлургического завода и показания температурных датчиков интеллектуальной системы управления отоплением зданий ЮУрГУ. Поиск диссонансов выполнен с помощью ранее разработанного автором статьи параллель-

ного алгоритма PD3 (Parallel DRAG-based Discord Discovery) [8] для графического процессора. Для визуализации найденных аномалий предложен метод построения тепловой карты диссонансов, имеющих различные длины, и алгоритм TopInterestDiscords, позволяющий выбрать в построенной тепловой карте наиболее значимые диссонансы независимо от их длин. Алгоритм подбирает диссонансы, которые имеют бóльшую оценку аномальности (независимо от длин диссонансов), исключая при этом пересекающиеся диссонансы. Для каждого рассмотренного случая представлены тепловая карта найденных диссонансов и наиболее значимые из них.

Работа выполнена при финансовой поддержке Российского научного фонда (грант № 23-21-00465).

Литература

1. Blázquez-García A., Conde A., Mori U., Lozano J.A. A Review on Outlier/Anomaly Detection in Time Series Data // ACM Comput. Surv. 2021. Vol. 54, no. 3. 56:1–56:33. DOI: 10.1145/3444690.
2. Kumar S., Tiwari P., Zymbler M.L. Internet of Things is a revolutionary approach for future technology enhancement: a review // J. Big Data. 2019. Vol. 6. P. 111. DOI: 10.1186/s40537-019-0268-2.
3. Цымблер М.Л., Краева Я.А., Латыпова Е.А. и др. Очистка сенсорных данных в интеллектуальных системах управления отоплением зданий // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2021. Т. 10, № 3. С. 16–36. DOI: 10.14529/cmse210302.
4. Иванов С.А., Никольская К.Ю., Радченко Г.И. и др. Концепция построения цифрового двойника города // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2020. Т. 9, № 4. С. 5–23. DOI: 10.14529/cmse200401.
5. Keogh E.J., Lin J., Fu A.W. HOT SAX: efficiently finding the most unusual time series subsequence // Proceedings of the 5th IEEE International Conference on Data Mining (ICDM 2005), Houston, Texas, USA, November 27-30, 2005. IEEE Computer Society, 2005. P. 226–233. DOI: 10.1109/ICDM.2005.79.
6. Yankov D., Keogh E.J., Rebbapragada U. Disk aware discord discovery: Finding unusual time series in terabyte sized datasets // Proceedings of the 7th IEEE International Conference on Data Mining (ICDM 2007), October 28-31, 2007, Omaha, Nebraska, USA. 2007. P. 381–390. DOI: 10.1109/ICDM.2007.61.
7. Chandola V., Cheboli D., Kumar V. Detecting anomalies in a time series database. Retrieved from the University of Minnesota Digital Conservancy. 2009. URL: https://hdl.handle.net/11299/215791 (дата обращения: 12.04.2022).
8. Kraeva Y., Zymbler M. A parallel discord discovery algorithm for a graphics processor // Pattern Recognition and Image Analysis. 2023. Vol. 33, no. 2. P. 101–113. DOI: 10.1134/S1054661823020062.
9. Mueen A., Nath S., Liu J. Fast approximate correlation for massive time-series data // Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2010, Indianapolis, Indiana, USA, June 6-10, 2010. ACM, 2010. P. 171–182. DOI: 10.1145/1807167.1807188.

10. Han Z., Gao P., Wan F. Research on Data Mining and Visualization Technology // CONF-CDS 2021: The 2nd International Conference on Computing and Data Science, Stanford, CA, USA, January 28-30, 2021. ACM, 2021. 71:1–71:4. DOI: 10.1145/3448734.3450801.

11. Yeh C.M., Zhu Y., Ulanova L., *et al.* Time series joins, motifs, discords and shapelets: A unifying view that exploits the matrix profile // Data Min. Knowl. Discov. 2018. Vol. 32, no. 1. P. 83–123. DOI: 10.1007/s10618-017-0519-9.

12. Zimmerman Z., Kamgar K., Senobari N.S., *et al.* Matrix Profile XIV: Scaling Time Series Motif Discovery with GPUs to Break a Quintillion Pairwise Comparisons a Day and Beyond // Proceedings of the ACM Symposium on Cloud Computing, SoCC 2019, Santa Cruz, CA, USA, November 20-23, 2019. ACM, 2019. P. 74–86. DOI: 10.1145/3357223.3362721.

13. Chen X., Chen Y., He Z. Urban Traffic Speed Dataset of Guangzhou, China. 2018. DOI: 10.5281/zenodo.1205229.

14. Биленко Р.В., Долганина Н.Ю., Иванова Е.В., Рекачинский А.И. Высокопроизводительные вычислительные ресурсы Южно-Уральского государственного университета // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2022. Т. 11, № 1. С. 15–30. DOI: 10.14529/cmse220102.

15. Rosenthal D. CVC technology on hot and cold strip rolling mills // Rev. Met. Paris. 1988. Vol. 85, no. 7. P. 597–606. DOI: 10.1051/metal/198885070597.

16. Басалаев А.А. Автоматизированный энергоменеджмент теплоэнергетического комплекса университетского городка // Вестник ЮУрГУ. Серия: Компьютерные технологии, управление, радиоэлектроника. 2015. Т. 15, № 4. С. 26–32. DOI: 10.14529/ctcr150403.

Краева Яна Александровна, старший преподаватель, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

DOI: 10.14529/cmse230202

ANOMALY DETECTION IN DIGITAL INDUSTRY SENSOR DATA USING PARALLEL COMPUTING

© 2023 Ya.A. Kraeva

South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia)

E-mail: kraevaya@susu.ru

Received: 20.09.2022

The article presents the results of case studies on the anomaly discovery in sensor data from various applications of the digital industry. The time series data obtained from the sensors installed on machine parts and metallurgical equipment, and from the temperature sensors in the smart building heating control system are considered. The anomalies discovered in such data indicate an abnormal situation or failures in the technological equipment. In this study, the anomaly is formalized as a range discord, namely a subsequence, the distance from which to its nearest neighbor is not less than the threshold prespecified by an analyst. The nearest neighbor of the given subsequence is a subsequence that does not overlap with this one and has a minimum distance to it. The discord discovery is performed through the parallel algorithm for GPU developed by the author. To visualize the anomalies found, a discord heatmap method and an algorithm for selection the most interesting discords regardless of their lengths are proposed.

Keywords: time series, sensor data, anomaly detection, discord, parallel algorithm, GPU, CUDA.

FOR CITATION

Kraeva Ya.A. Anomaly Detection in Digital Industry Sensor Data Using Parallel Computing. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2023. Vol. 12, no. 2. P. 47–61. (in Russian) DOI: 10.14529/cmse230202.

References

1. Blázquez-García A., Conde A., Mori U., Lozano J.A. A Review on Outlier/Anomaly Detection in Time Series Data. ACM Comput. Surv. 2021. Vol. 54, no. 3. 56:1–56:33. DOI: 10.1145/3444690.
2. Kumar S., Tiwari P., Zymbler M.L. Internet of Things is a revolutionary approach for future technology enhancement: a review. J. Big Data. 2019. Vol. 6. P. 111. DOI: 10.1186/s40537-019-0268-2.
3. Zymbler M.L., Kraeva Y.A., Latypova E.A., *et al.* Cleaning Sensor Data in Intelligent Heating Control System. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2021. Vol. 10, no. 3. P. 16–36. (in Russian) DOI: 10.14529/cmse210302.
4. Ivanov S.A., Nikolskaya K.Y., Radchenko G.I., *et al.* Digital Twin of a City: Concept Overview. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2020. Vol. 9, no. 4. P. 5–23. (in Russian) DOI: 10.14529/cmse200401.
5. Keogh E.J., Lin J., Fu A.W. HOT SAX: efficiently finding the most unusual time series subsequence. Proceedings of the 5th IEEE International Conference on Data Mining (ICDM

2005), Houston, Texas, USA, November 27-30, 2005. IEEE Computer Society, 2005. P. 226–233. DOI: 10.1109/ICDM.2005.79.

6. Yankov D., Keogh E.J., Rebbapragada U. Disk aware discord discovery: Finding unusual time series in terabyte sized datasets. Proceedings of the 7th IEEE International Conference on Data Mining (ICDM 2007), October 28-31, 2007, Omaha, Nebraska, USA. 2007. P. 381–390. DOI: 10.1109/ICDM.2007.61.
7. Chandola V., Cheboli D., Kumar V. Detecting anomalies in a time series database. Retrieved from the University of Minnesota Digital Conservancy. 2009. URL: https://hdl.handle.net/11299/215791 (accessed: 12.04.2022).
8. Kraeva Y., Zymbler M. A parallel discord discovery algorithm for a graphics processor. Pattern Recognition and Image Analysis. 2023. Vol. 33, no. 2. P. 101–113. DOI: 10.1134/S1054661823020062.
9. Mueen A., Nath S., Liu J. Fast approximate correlation for massive time-series data. Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2010, Indianapolis, Indiana, USA, June 6-10, 2010. ACM, 2010. P. 171–182. DOI: 10.1145/1807167.1807188.
10. Han Z., Gao P., Wan F. Research on Data Mining and Visualization Technology. CONFCDS 2021: The 2nd International Conference on Computing and Data Science, Stanford, CA, USA, January 28-30, 2021. ACM, 2021. 71:1–71:4. DOI: 10.1145/3448734.3450801.
11. Yeh C.M., Zhu Y., Ulanova L., *et al.* Time series joins, motifs, discords and shapelets: A unifying view that exploits the matrix profile. Data Min. Knowl. Discov. 2018. Vol. 32, no. 1. P. 83–123. DOI: 10.1007/s10618-017-0519-9.
12. Zimmerman Z., Kamgar K., Senobari N.S., *et al.* Matrix Profile XIV: Scaling Time Series Motif Discovery with GPUs to Break a Quintillion Pairwise Comparisons a Day and Beyond. Proceedings of the ACM Symposium on Cloud Computing, SoCC 2019, Santa Cruz, CA, USA, November 20-23, 2019. ACM, 2019. P. 74–86. DOI: 10.1145/3357223.3362721.
13. Chen X., Chen Y., He Z. Urban Traffic Speed Dataset of Guangzhou, China. 2018. DOI: 10.5281/zenodo.1205229.
14. Bilenko R.V., Dolganina N.Y., Ivanova E.V., Rekachinsky A.I. High-performance Computing Resources of South Ural State University. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2022. Vol. 11, no. 1. P. 15–30. (in Russian) DOI: 10.14529/cmse220102.
15. Rosenthal D. CVC Technology on Hot and Cold Strip Rolling Mills. Rev. Met. Paris. 1988. Vol. 85, no. 7. P. 597–606. DOI: 10.1051/metal/198885070597.
16. Basalaev A.A. Automated Energy Management for Heat and Power System of University Campus. Bulletin of the South Ural State University. Series: Computer Technologies, Automatic Control, Radio Electronics. 2015. Vol. 15, no. 4. P. 26–32. (in Russian) DOI: 10.14529/ctcr150403.