DOI 10.37882/2223-2966.2023.9-2.23

РАСПРЕДЕЛЕННАЯ МОДЕЛЬ ВЫЧИСЛЕНИЙ MAP REDUCE ДЛЯ R/S АНАЛИЗА ВРЕМЕННЫХ РЯДОВ БОЛЬШОЙ РАЗМЕРНОСТИ

MAP REDUCE FOR R/S ANALYSIS OF HIGH-DIMENSIONAL TIME SERIES

A. Sabutkevich
D. Vikhlyaev
I. Nikiforov

Summary. The work is devoted to research approaches to improve the efficiency of computing indicators of high-dimensional time series, presented using ungrouped streaming data, using the example of using R/S analysis. The proposed method is based on the use of the distributed computing model Map Reduce for the implementation of the R/S analysis algorithm. The proposed solution is implemented in the software tool, which makes it possible to increase the efficiency of calculations using the correct cluster configuration by an average of 34 % compared to the traditional computing method for experimental dataset.

Keywords: distributed computing, Big Data, Map Reduce, R/S analysis, Hurst exponent, time series.

Сабуткевич Артем Михайлович
Санкт-Петербургский политехнический университет Петра Великого
artem.sabut@gmail.com
Вихляев Дмитрий Александрович
Санкт-Петербургский политехнический университет Петра Великого
dim49v@yandex.ru
Никифоров Игорь Валерьевич
кандидат техн. наук, доцент, Санкт-Петербургский политехнический университет Петра Великого
nikiforov_iv@spbstu.ru

Аннотация. Работа посвящена исследованию подходов повышения эффективности вычислений показателей временных рядов большой размерности, представленных при помощи не сгруппированных потоковых данных, на примере применения R/S анализа. Предложен метод, основывающийся на использовании модели распределенных вычислений Map Reduce для реализации алгоритма R/S анализа. Предложенное решение реализовано в программном средстве, применение которого позволило повысить эффективность проводимых вычислений при использовании корректной конфигурации кластера в среднем на 34 % по сравнению с методом традиционных вычислений для экспериментальных данных.

Ключевые слова: распределенные вычисление, большие данные, Map Reduce, R/S анализ, экспонента Херста, временные ряды.

Введение

Анализ экспериментальных или исторических данных с целью выявления закономерностей является актуальной задачей во многих научных и прикладных областях.

Нередко исходные данные могут быть представлены при помощи временных рядов, что позволяет использовать для их анализа совокупность математико-статистических методов. Областями, в которых активно применяется анализ временных рядов, являются экономика [1], социология, промышленность, машиностроение, информационные технологии и другие.

Соответствующие временные ряды могут обладать большой размерностью, что затрудняет или делает практически невозможным проведение их теоретического анализа. Для уменьшения трудоемкости данного процесса используется программно-вычислительные комплексы [2]. Однако классические подходы организации вычислений могут оказываться недостаточно эффективными в связи с большим объемом данных. В связи с этим особую актуальность имеет повышение эффективности данного процесса [3] за счет применения распределенного характера вычислений.

В рамках данной работы рассматривается подход к применению распределенных вычисления для временных рядов большой размерности при помощи модели Map Reduce в области экономики. В качестве вычисляемого значения используется экспонента Херста [4], а временной ряд описывает изменение стоимости акций на фондовом рынке. Данные временного ряда имеют потоковый характер [5] с отсутствием группировки относительно наименований акций.

Обзор литературы

Согласно центральной предельной теореме — при увеличении числа испытаний предельное распределение случайной системы будет близким к нормальному распределению [6]. При этом все события должны быть независимыми и идентично распределены. В процессе исследования множества сложных систем обычно предполагают гипотезу о нормальности системы, чтобы к ней можно было применить классический статический анализ.

На практике многие системы, в том числе финансовые рынки и соответствующие им временные ряды, не являются нормально-распределенными. Именно для анализа таких систем применяется *R*/*S* анализ [7]. Данный подход позволяет различить случайный и фрактальный временные ряды [8], а также делать выводы о наличии непериодических циклов, долговременной памяти и других характеристик [9].

Алгоритм *R*/*S* анализа задается следующей последовательностью действий [10] [11]:

1. Для исходного ряда S_t определяется логарифмическое отношение:

$$N_t = \ln\frac{S_t}{S_{t-1}}.$$

2. Ряд *N* разделяется на *A* смежных периодов длиной *n*. Каждый период определяется как I_a, где $a = 1, 2. \ldots, A$. Далее определяется среднее значение для каждого I_a:

$$E(I_a) = \frac{1}{n}\sum_{k=1}^{n} N_{k,a}.$$

3. Вычисляются отклонения от среднего значения для каждого периода I_a:

$$X_{k,a} = \sum_{i=1}^{k}\left(N_{i,a} - E(I_a)\right).$$

4. Вычисляется размах в пределах каждого периода:

$$R_{I_a} = \max(X_{k,a}) - \min(X_{k,a}).$$

5. Вычисляется стандартное отклонение для каждого периода I_a:

$$S_{I_a} = \sqrt{\frac{1}{n}\sum_{k=1}^{n}\left(N_{k,a} - E(I_a)\right)^2}.$$

6. Каждый R_{I_a} делится на соответствующее значение S_{I_a}. На основе этих данных рассчитывается среднее значение *R*/*S*:

$$R / S(n) = \frac{\sum_{a=1}^{A} R / S(A)}{A}.$$

7. Шаги 2–6 повторяются пока $n < \frac{N}{2}$. На каждом шаге увеличивается значение *n*.

8. Далее строится график зависимости $\log(R/S(n))$ от $\log(n)$. При помощи метода наименьших квадратов находится регрессия вида $\log(R/S(n)) = H * \log(n) + c$, где *H* является экспонентой Херста.

На основе полученного значения экспоненты Херста можно сделать вывод о характеристики временного ряда:

- $H < 0{,}5$ — анти-персистентный ряд: за высоким значением следует низкое значение и наоборот;
- $H = 0{,}5$ — явная тенденция не выражена;
- $H < 0{,}5$ — персистентный ряд: за высоким значением следует более высокое значение и наоборот.

Существующие решения

Существующие подходы и реализации можно классифицировать следующим образом:

- обособленные программные решения;
- библиотеки и модули для использования в составе различных программных средств.

Основными критериями, выделяемыми при проведении сравнительного анализа существующих подходов и реализаций (см. табл. 1), являются:

- поддержка модели распределенных вычислений для большого объема данных;
- возможность расширения функционала путем интеграции дополнительных модулей обработки данных;
- реализация на языке высокого уровня или наличие API;
- поддержка обработки котировок для различных акций, представленных в виде потоковых данных с отсутствием группировки.

На основе приведенных данных можно сделать вывод, что среди рассмотренных решений не существует подхода, удовлетворяющего всем критериям, поэтому

Таблица 1.

Сравнительный анализ существующих решений

Название	Поддержка распределенных вычислений	Возможность расширения функционала	Реализация на языке высокого уровня	Поддержка обработки потоковых котировок
Программный комплекс «Симметрия» [12]	–	–	+–	–
Приложение «STATA» [13]	–	+	+	–
Подход, рассмотренный в работе [14]	–	+	+–	–
Реализация R/S анализа с использованием Matlab [15]	–	+	+	–
Методика, описанная в работе [16]	+	+–	+	–

актуальной является задача разработки собственного программного решения.

Предлагаемый подход

Модель данных

Модель данных включает в себя информацию о ключевых характеристиках биржевой котировки, необходимых для применения R/S анализа, а именно: наименовании акции (строковый тип данных), временной характеристики периода (строковый тип данных, удовлетворяющий паттерну задания даты и времени) и соответствующей стоимости одной акции на конец периода (вещественный тип данных).

Компонент обработки данных

На вход в компонент обработки данных подаются биржевые котировки для различных акций, а выходными значениями являются экспоненты Херста, вычисленные для каждой из исходных акций.

Основным элементом компонента обработки данных является инструмент анализа данных, в качестве которого используется платформа Apache Hadoop [17]. В ее основе лежит распределенный подход к вычислению и хранению информации [18]. Выбор обуславливается поддержкой обработки больших объемов данных, а также легкостью данного инструмента. Другими важными преимуществами являются эффективность, обеспечиваемая использованием модели распределенных данных, а также высокая надежность за счет возможности хранения нескольких копий данных.

Также был рассмотрен Apache Spark [17], но его использование связано с дополнительными ограничениями для ОЗУ и слишком высокой сложностью конфигурирования. На основе проведенной теоретической оценки использование Spark может оказать более сильное влияние на повышение эффективности вычислений, однако описанные недостатки являются более значимыми.

На программном уровне Hadoop представляет собой программный фреймворк, позволяющий хранить данные при помощи распределённой файловой системы Hadoop Distributed File System [19] и обрабатывать их с использованием вычислительных кластеров на основе модели Map Reduce [20]. В соответствии с подходом Map Reduce обработка данных состоит из шагов: Map, Shuffle и Reduce [21]. В качестве ключа, применяемого для создания пары на шаге Map, используется наименование акции, а соответствующим значением является ее стоимость. Вычисление экспоненты Херста осуществляется на этапе Reduce. Демонстрационный пример предлагаемой модели вычислений в одной из возможных конфигураций представлен на рис. 1.

Модель может иметь различные конфигурации в зависимости от количества узлов кластера.

Реализация программного инструмента

Программное решение представляет собой дистрибутив, разработанный для ОС Linux, с поддержкой автоматизации развертывания, запуска Hadoop с предварительной перезагрузкой используемых узлов и запуска выполнения конкретной Map Reduce задачи.

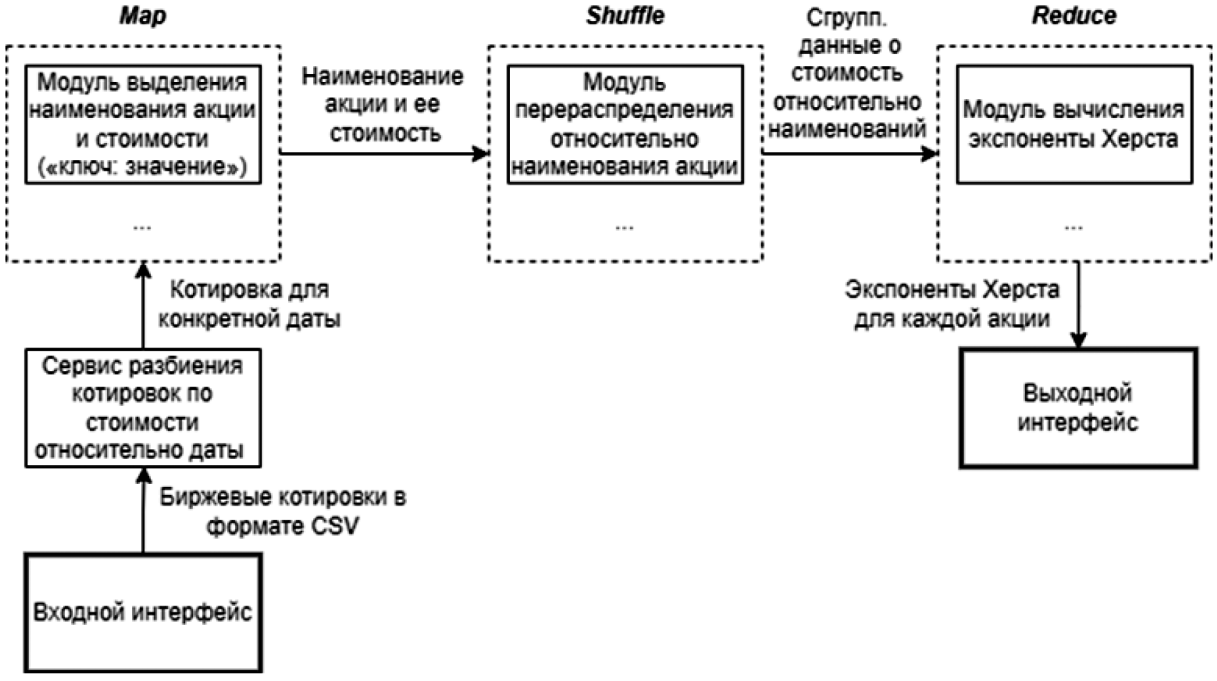

Рис. 1. Демонстрационный пример модели вычислений

В качестве основного языка программирования использован Python версии 3.8.10. Выбор объясняется наличием множества дополнительных модулей и библиотек, упрощающих процесс реализации, а также его поддержкой всеми средствами и технологиями, используемыми при разработке.

Реализация компонента обработки данных

Для реализации системы была использована версия Apache Hadoop 1.0.3 по причине относительной простоты автоматизации развертывания на узлах кластера. В ходе проведенных тестирований, перегрузки, свойственные данной версии, выявлены не были.

Для обеспечения функциональной масштабируемости компонент обработки данных поддерживает возможность вычисления различных показателей и метрик для исходных данных посредством разработки дополнительных модулей. Основным ограничением для них является необходимость реализации статического класса, содержащего публичный метод для вычисления целевой метрики. Параметром указанного метода является список, хранящий значения временного ряда.

Реализация алгоритма R/S анализа

Реализация алгоритма R/S анализа представлена посредством отдельного модуля с использованием библиотеки Numpy версии 1.12.4, реализующей все необходимые вычислительные методы и структуры хранения данных:

— array — массив, используемый для хранения временных рядов.
— subtract — метод вычисления разность между элементами массивов.
— std — метод вычисления среднеквадратического отклонения.
— polyfit — метод наименьших квадратов.
— log — метод вычисление логарифма для элементов массива.

Реализация модели Map Reduce

Шаги Map и Reduce реализованы посредством одноименных модулей без использования дополнительных внешних библиотек. Реализация Reduce задействует модуль алгоритма R/S анализа, описанный раннее. Операция Shuffle, в свою очередь, производится автоматически и не требует реализации.

Создание кластера для распределенной работы

Главный сервер (master) и все зависимые сервера (slave) расположены на отдельных вычислительных машинах. Объединение их в кластер реализуется при помощи SSH ключей, которые формируются при настройке конфигурации системы. Для главного сервера в конфигурации должны быть указан все зависимые сервера. Средствами главного сервера осуществляется запуск конкретной Map Reduce задачи.

Эксперимент

Кластер, при помощи которого проводился эксперимент, был реализован на основе нескольких серверов при помощи сервиса Digital Ocean. Узлами кластера являлись сформированные виртуальные машины на базе Linux (droplet) с характеристиками 1 ядро 3.3 ГГц, ОЗУ 1 Гб. Дополнительно был задействован персональный компьютер, имеющие следующие характеристики: 4 ядра 3.6 ГГц, ОЗУ 16 Гб.

Для проведения исследования были подготовлены искусственные наборы данных заданной размерности: 450 Мб, 930 Мб, 1830 Мб и 2850 Мб. Данный подход допустим, так как время вычислений не зависит от конкретных значений стоимости акции. В зависимости от размерности данных использовались различные размеры блоков, определяющие длину периодов, на которые разделяется временной ряд.

Использование одного узла кластера

В табл. 2 приведены значения времени, затраченного на осуществления вычислений, относительно зависимости размера набора данных и характера вычислений как с использованием Map Reduce модели, так и при помощи традиционных вычислений. На основе приведенных данных построен график (см. рис. 2).

На основе полученных данных можно сделать вывод, что наиболее эффективным по времени оказалось использование традиционных вычислений без применения Map Reduce модели.

Таблица 2.

Сравнение времени обработки данных на одном узле

Размер набора данных		450 Мб	930 Мб	1830 Мб	2850 Мб
Reducer	Node				
1	1	160 сек.	316 сек.	616 сек.	936 сек.
Персональный компьютер		139 сек.	260 сек.	456 сек.	727 сек.
Количество блоков		3	6	10	15

В среднем применение вычислений с использованием персонального компьютера оказывается на 25 % эффективнее по времени, чем при помощи модели Map Reduce при конфигурации на одном узле.

Это обуславливается тем, что Hadoop и, в частности, Map Reduce модель, оказываются более эффективными при использовании горизонтального масштабирования.

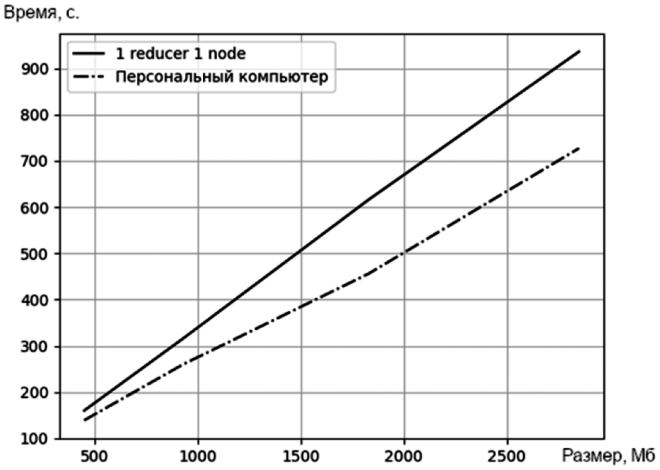

Рис. 2. График времени обработки данных на одном узле

Использование нескольких узлов кластера

Для исследования времени работы на нескольких узлах были дополнительно сформированы конфигурации с 2 узлами, отличающиеся количеством reducer. Сводные данные представлены в табл. 3, а соответствующий график на рис. 3.

Таблица 3.

Сравнение времени обработки данных на нескольких узлах

Размер набора данных		450 Мб	930 Мб	1830 Мб	2850 Мб
Reducer	Node				
2	2	118 сек.	185 сек.	357 сек.	487 сек.
1	2	175 сек.	245 сек.	557 сек.	853 сек.
1	1	160 сек.	316 сек.	616 сек.	936 сек.
Персональный компьютер		139 сек.	260 сек.	456 сек.	727 сек.
Количество блоков		3	6	10	15

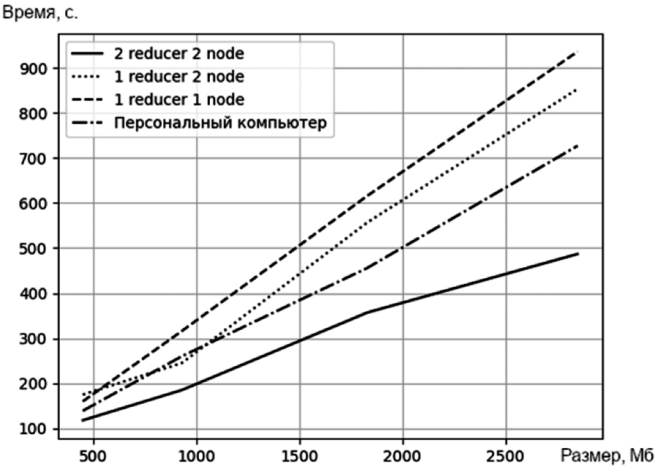

Рис. 3. График времени обработки данных на нескольких узлах

Из приведенных данных можно сделать вывод, что для заданных данных при увеличении количества узлов кластера и количества reducer скорость вычислений при использовании Hadoop и модели Map Reduce возрастает.

При использовании конфигурации с 2 узлами и 1 reducer скорость обработки данных оказывается выше, чем при использовании только 1 узла и соответственно 1 reducer. А, в свою очередь, конфигурация с 2 узлами и 2 reducer по времени показывает наилучший результат среди всех рассмотренных конфигураций, в среднем превышающий на 34 % время обработки с использованием персонального компьютера.

Заключение

В работе предложен подход вычисления экспоненты Херста для целевых котировок фондового рынка, формально представленных в виде временного ряда большой размерности, посредством R/S анализа.

В ходе применения предложенного подхода при корректном конфигурировании кластера удалось достичь повышения эффективность в среднем на 34% в сравнении с использованием модели локальных вычислений для исходных экспериментальных данных различной размерности.

Оценка эффективности была произведена при помощи программной реализации предложенного подхода с использованием Apache Hadoop 1.0.3.

Ключевыми отличительными особенностями предложенного подхода являются:

— использование модели распределенных вычислений Map Reduce, обеспечивающий повышение эффективности вычислений и возможности обработки временных рядов большой размерности;
— поддержка функциональной масштабируемости, а также интеграции в другие системы;
— наличие возможности обработки данных потокового характера, не сгруппированных относительно конкретной целевой акции.

ЛИТЕРАТУРА

1. Михайличенко, А.А. Анализ временных рядов и методы прогнозирования в современной экономике / А.А. Михайличенко, Ю.В. Кольцов // Машиностроение: Сборник научных статей / ГОУ ВПО КубГТУ, ООО «Издательский Дом — Юг». Том Выпуск 3. — Краснодар: Общество с ограниченной ответственностью «Издательский Дом — Юг», 2009. — С. 67–70. — EDN TCKRBF.
2. Тартаковский, В.А. Разработка программных средств для обработки больших временных рядов / В.А. Тартаковский, И.А. Ботыгин, А.И. Шерстнева // Девятая Сибирская конференция по параллельным и высокопроизводительным вычислениям: Сборник статей, Томск, 10–12 октября 2017 года / под редакцией А.В. Старченко. — Томск: Национальный исследовательский Томский государственный университет, 2017. — С. 105–109. — DOI 10.17223/9785946216531/15. — EDN DKTDKI.
3. Кирилюк, И.Л. Оптимизация сложности моделей анализа временных рядов в экономике / И.Л. Кирилюк, А.В. Кузнецова, О.В. Сенько // Математические методы распознавания образов ММРО-2017 : тезисы докладов 18-й Всероссийской конференции с международным участием, Таганрог, 09 октября 2017 года — 13 2019 года. — Москва: Общество с ограниченной ответственностью «ТОРУС ПРЕСС», 2017. — С. 64–65. — EDN BHAPPX.
4. Кутузов, А.В. Оценки интервалов квазистационарности временных рядов экономических показателей на основе их мультифрактальных моделей / А.В. Кутузов, А.А. Иванков // Технологическая перспектива в рамках Евразийского пространства: новые рынки и точки экономического роста : Материалы 2-й Международной конференции, Санкт-Петербург, 20–22 октября 2016 года. — Санкт-Петербург: Центр научно-информационных технологий «Астерион», 2016. — С. 186–189. — EDN XVNCLZ.
5. Конфигурируемая система сбора и обработки потоковых данных на основе SAP HANA / В.В. Монастырев, А.В. Назаров, А.М. Акимов [и др.] // Информатика и кибернетика (ComCon-2017): Сборник докладов студенческой научной конференции Института компьютерных наук и технологий, Санкт-Петербург, 03–08 апреля 2017 года. — Санкт-Петербург: Федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский политехнический университет Петра Великого», 2017. — С. 354–358. — EDN XOZWDB.
6. Ивченко, Г.И. Математическая статистика: Учебник. / Г.И. Ивченко, Ю.И. Медведев. — М.: Высшая школа, 1984. — 248 с.
7. Херст, Г.Э. Долгосрочная вместимость водохранилищ. Труды Американского общества гражданских инженеров, 1951. — Т. 116. — С. 770.
8. Червова, А.А. Фрактальный анализ нестационарных временных рядов различной природы / А.А. Червова // Естественные и технические науки. — 2015. — № 11(89). — С. 408–412. — EDN VHTLOH.
9. Гузикова, Л.А. Опыт фрактального анализа цен акций российских компаний / Л.А. Гузикова, Н.М. Молодежев // Современные аспекты экономики. — 2020. — № 5-2(273). — С. 100–112. — EDN CIGWDL.
10. Гачков, А.А. Рандомизированный алгоритм R/S-анализа финансовых рядов / А.А. Гачков // Стахостическая оптимизация в информатике / Под ред. О.Н. Граничина. — 2009. — № 5. — С. 40–64.
11. Зиненко, А.В. R/S анализ на фондовом рынке / А.В. Зиненко // Бизнес-информатика. — 2012. — № 3 (21). — С. 24–30.
12. Пимонов, И.А. Комплекс программ для оценки и анализа фрактальных свойств фондового рынка / И.А. Пимонов, А.И. Трегуб // Информационные технологии. — 2008. –№4. — С. 105–110.
13. Баум, К.Ф. Эконометрика. Применение пакета Stata: Учебник и практикум / К.Ф. Баум, Г.И. Пеникас, С.А. Айвазян. — М: Общество с ограниченной ответственностью «Издательство ЮРАЙТ», 2016. — 352 с. — ISBN 978-5-9916-6993-1. — EDN VTWXYP.
14. Масловская, А.Г. Применение фрактальных методов для анализа динамических данных / А.Г. Масловская, Т.Р. Осокина, Т.К. Барабаш // Вестник Амурского государственного университета. — 2010. — №51: Серия: Естественные и экономические науки. — С. 13–20.
15. Теплов, С.Е. Применение R/S-анализа на фондовых рынках / С.Е. Теплов // Финансы и бизнес. — 2008. — № 1. — С. 129–137.
16. Kussainov, A.S. Hurst exponent estimation, verification, portability and parallelization / A.S. Kussainov, S.G. Kussainov // Recent Contibutions to Physics. — 2015. — № 1(52). — P. 98–103. — EDN XKOKWH.
17. Система обработки больших данных для анализа событий репозитория GitHub / Н.В. Воинов, К Родригес Гарсон, И.В. Никифоров, П.Д. Дробинцев // Международная конференция по мягким вычислениям и измерениям. — 2019. — Т. 1. — С. 283–286. — EDN TSGWMD.
18. Картанова, А.Д. Инструмент Hadoop и оптимизация хранилища данных / А.Д. Картанова, А.Б. Абдрасакова, Т.И. Иманбеков // Современные проблемы механики. — 2020. — № 42(4). — С. 67–82. — EDN CHFRKG.
19. Choi, W. Gi. A write-friendly approach to manage namespace of Hadoop distributed file system by utilizing nonvolatile memory / W. Gi. Choi, S. Park // The Journal of Supercomputing. — 2019. — Vol. 75, № 10. — P. 6632–6662. — DOI 10.1007/s11227-019-02876-9. — EDN BTBRLF.
20. Никифоров, И.В. Курсовое проектирование по учебной дисциплине «Наука о данных и аналитика больших объемов информации»: Учебное пособие / И.В. Никифоров. — Санкт-Петербург: Федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский политехнический университет Петра Великого», 2017. — 62 с. — ISBN 978-5-7422-5638-0. — EDN XPRQXB.
21. Гладкий, М.В. Модель распределенных вычислений MapReduce / М.В. Гладкий // Труды БГТУ. №6. Физико-математические науки и информатика. — 2016. — № 6(188). — С. 194–198. — EDN XAGKHD.