ISSN 2500-316X (Online)

https://doi.org/10.32362/2500-316X-2019-7-6-56-67

UDC 004.75

Processing streams in a monitoring cloud cluster

Alexey N. Nazarov

MIREA – Russian Technological University, Moscow 119454, Russia
@Corresponding author, e-mail: a.nazarov06@bk.ru

The creation of monitoring clusters based on cloud computing technologies is a promising direction for the development of systems for continuous monitoring of objects for various purposes in the web space. Hadoop web-programming environment is the technological basis for the development of algorithmic and software solutions for the synthesis of monitoring clusters, including information security and information counteraction systems. The International Telecommunication Union' (ITU) recommendations Y. 3510 present the requirements for cloud infrastructure that require monitoring the performance of deployed applications based on the collection of real-world statistics. Often, computing resources of monitoring clusters of cloud data centers are allocated for continuous parallel processing of high-speed streaming data, which imposes new requirements to monitoring technologies, necessitating the creation and research of new models of parallel computing. The need to use service monitoring plays an important role in the cloud computing industry, especially for SLA/QoS assessment, as the application or service may experience problems even if the virtual machines on which the work is taking place appear to be operational. This requires to study the methodological possibilities of organization to study of parallel processing high-speed streaming services with the processing of huge amounts of bit data, and, simultaneously, to estimate the necessary computational resource. In the conditions of high dynamics of changes in the bit rate of information generation from the source, a model of the bit rate of Discretized Stream (DStream) formation is proposed, which has a common application. Based on the poly-burst nature of the bit rate model, a model of group content traffic of any sources of different services processed in the cloud cluster was created. The obtained results made it possible to develop mathematical models of parallel DStreams from sources processed in a cloud cluster via Hadoop technology using the micro-batch architecture of the Spark Streaming module. These models take into account the flow of requests for maintenance from sources of different services, on the one hand, and, on the other hand, the needs of services in bit rate, taking into account the multichannel traffic of sources of various services. At the same time, analytical relations are obtained to calculate the required performance of the Hadoop cluster at a given value of the probability of batch loss.

Keywords: monitoring, Hadoop, Spark, batch, bit rate, micro-batch, architecture, parallel flow, cloud computing, expectation, variance, probability, probability distribution function, probability distribution density, random process, Delta function.

For citation: Nazarov A.N. Processing streams in a monitoring cloud cluster. *Rossiiskii tekhnologicheskii zhurnal* = Russian Technological Journal. 2019;7(6):56-67. https://doi.org/10.32362/2500-316X-2019-7-6-56-67

Обработка потоков в мониторинговом облачном кластере

А.Н. Назаров

МИРЭА – Российский технологический университет, Москва 119454, Россия
@Автор для переписки, e-mail: a.nazarov06@bk.ru

Создание мониторинговых кластеров облачных вычислений является перспективным направлением создания систем непрерывного контроля объектов различного назначения в web-пространстве. Среда web-программирования Hadoop является технологической основой разработки алгоритмических и программных решений по синтезу мониторинговых кластеров, включая системы информационной безопасности и информационного противодействия. В рекомендациях Y.3510 Международного союза электросвязи (ITU) представлены требования, предъявляемые к облачной инфраструктуре, обусловливающие необходимость в мониторинге производительности развернутых приложений на основе сбора реальных статистических данных. Зачастую вычислительные ресурсы мониторинговых кластеров облачных центров обработки данных выделены для постоянной параллельной обработки высокоскоростных потоковых данных, что предъявляет новые требования к технологиям мониторинга, обусловливающие необходимость создания и исследования новых моделей параллельных вычислений. Необходимость применения мониторинга услуг играет важную роль в индустрии облачных вычислений, в особенности для оценки SLA/QoS, так как в приложении или услуге могут возникнуть проблемы, даже если виртуальные машины, на которых происходит работа, выглядят работоспособными. При этом необходимо решение задачи исследования методических возможностей по оценке необходимого вычислительного ресурса в условиях высокоскоростных потоковых сервисов с обработкой гигантских объемов битовых данных. Разработаны математические модели параллельных потоков DStream от источников, обрабатываемых в облачном кластере на технологии Hadoop с использованием «микропакетной» архитектуры модуля Spark Streaming, учитывающие, с одной стороны, поток заявок источников различных сервисов на обслуживание, а, с другой стороны, потребности сервисов в битовой скорости передачи с учетом полипачечности трафика источников различных сервисов.

Ключевые слова: мониторинг, Hadoop, Spark, пакет, битовая скорость, микропакет, архитектура, параллельный поток, облачные вычислениямматематическое ожидание, дисперсия, вероятность, функция распределения вероятностей, плотность распределения вероятностей, случайный процесс, Дельта-функция.

Для цитирования: Nazarov A.N. Processing streams in a monitoring cloud cluster. *Российский технологический журнал.* 2019;7(6):56-67. https://doi.org/10.32362/2500-316X-2019-7-6-56-67

Introduction

The highest rates of use of Internet technologies and, above all, the Internet of things, in various areas of human activity, impose new requirements to ensure the control of various objects in the web space, including the interests of information security. Automation processes of industrial facilities with a continuous production cycle widely used technologies of industrial Internet and cloud monitoring. Intellectual analysis of the blogosphere is developing, which allows us to predict and synthesize political actions. Monitoring procedures are constantly functionally improved, including based on the latest technological solutions derived from the

achievements of artificial intelligence and cloud computing. Cyberattacks can cause enormous material and financial damage, especially in relation to critical information infrastructure.

Cloud computing is a dynamically scalable way to access external computing resources in the form of a service [2] provided via the Internet [3]. Efforts to standardize cloud technologies are consolidated by the International Telecommunication Union, whose work was initially carried out within the Focus Group on Cloud Computing. The results of this work were issued in the form of several reports [4, 5], and in the summer of 2013, specifications [6, 7] defining the requirements for the quality of cloud services, infrastructure and features of computing resources management were issued. As cloud computing technologies evolve, end-to-end and trusted management are of particular interest [8–12].

The requirements for cloud infrastructure include the need to monitor the performance of the data center, including parallel processing of streaming services.

The creation of a cloud monitoring cluster based on Hadoop technology allows solving many problems of continuous monitoring of objects in cyberspace.

Apache Spark is a versatile and high-performance cluster platform. Spark performance outperforms popular implementation of the MapReduce model, including streaming processing [13].

If Spark is installed on an existing Hadoop YARN cluster, it is possible to use the built-in managers of these clusters. RDD (Resilient Distributed Datasets) datasets are a collection of elements distributed among many computational nodes that can be processed in parallel. Spark Core provides many functions for managing such collections [13].

At the same time, it is necessary to solve the problem of studying methodological possibilities for assessing the necessary computing resource in the conditions of high-speed streaming services with the processing of huge volumes of bit data.

Concurrency in a cluster

At the logical level, RDD is a single collection of objects. During execution, RDD is divided into many partitions, each containing a subset of all the data. When Spark schedules and executes tasks, one task is created for each partition, and each task will run on the same core by default. In the Hadoop HDFS cluster, the original RDD sets are partitioned into HDFS file blocks.

If the degree of parallelism is not high enough, the Spark source resources may be idle. For example, if an application has 1000 cores at its disposal and it is performing a stage of only 30 tasks, you could increase the degree of parallelism and use more cores. In contrast, if the degree of parallelism is too high, the small overhead costs associated with each partition can be substantial in total.

The Spark Streaming module is built using a micro-batch architecture, when the data stream is interpreted as a continuous sequence of small data batches. Spark Streaming takes data from different sources and combines them into small batches (see Fig. 1) [13]. New batches are created at regular intervals. A new batch is created at the beginning of each time interval, and any data received during that interval is included in the batch. At the end of the interval, the batch increment stops. The size of the interval is determined by a parameter called the batch interval. Typically, the batch interval is between 500 milliseconds and a few seconds. Each batch generates a set of RDD and is processed by Spark job that creates another set of RDD. The batch processing results can then be transmitted to external systems for further analysis [13].

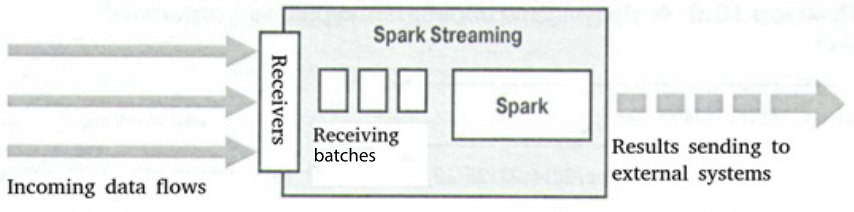

Fig. 1. Illustration of a micro-batch architecture.

Software abstraction in Spark Streaming is a discretized stream, or DStream, schematically represented in Fig. 2 as a sequence of RDD sets, where each RDD corresponds to one time interval.

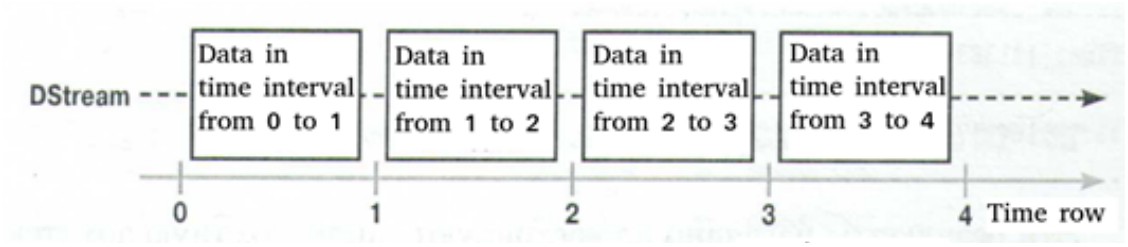

Fig. 2. Scheme of formation of a DStream.

Hadoop-based monitoring of objects in web space. Statement of the research problem

Monitoring of objects in the web-space implies regular observations of Internet objects (IP-addresses of users of the world wide web, sites, etc.), of their information and other resources, of services for both legal entities and individuals. These observations allow highlighting the state of these objects and the processes occurring in them under the influence of Internet activity on the Earth. Depending on the target function, web-monitoring also assesses the status and functional activities, the value of Internet ecosystems. Secondly conditions are created to determine corrective actions in cases where the targets of problem-oriented conditions are not achieved.

Using the ideas of artificial intelligence [14, 15] in combination with the capabilities of cloud computing for the development of monitoring technologies is very promising.

Hadoop, as a cloud technology of distributed processing of large amounts of data in the web environment, is rapidly becoming an important tool, a skill for a wide range of programmers [16].

In this regard, monitoring in the Hadoop environment will be understood as organized monitoring of selected objects in the web-space (in the subject area) using the capabilities of Hadoop.

Hadoop was created to work with Big Data in the web space. And in this regard it has a number of unique properties and abilities. Relevant quote [16]: “Technically speaking, Hadoop is an open source framework designed to create and run distributed applications that process large amounts of data.”

Hadoop operates on the basis of the MapReduce technology developed by Google. MapReduce is a simple but very powerful way to process and analyze very large datasets, and it is particularly effective for volumes ranging from a few petabytes.

In [17] the principles, approaches and technological procedures for the organization of monitoring are analyzed from rather general premises. Methodological approaches to the

creation of algorithms and software solutions in the Hadoop web-programming environment applied to a wide class of tasks for monitoring objects in the web-space were developed.

For the first time, the topology of the Hadoop monitoring cluster, which has a common application and is schematically shown in Fig. 3, was developed. Research was conducted, and algorithms were proposed for the measurement of attributes of monitored objects in the web space taking into account the requirements of measurements uniformity. System requirements for the design of the Hadoop monitoring cluster were developed.

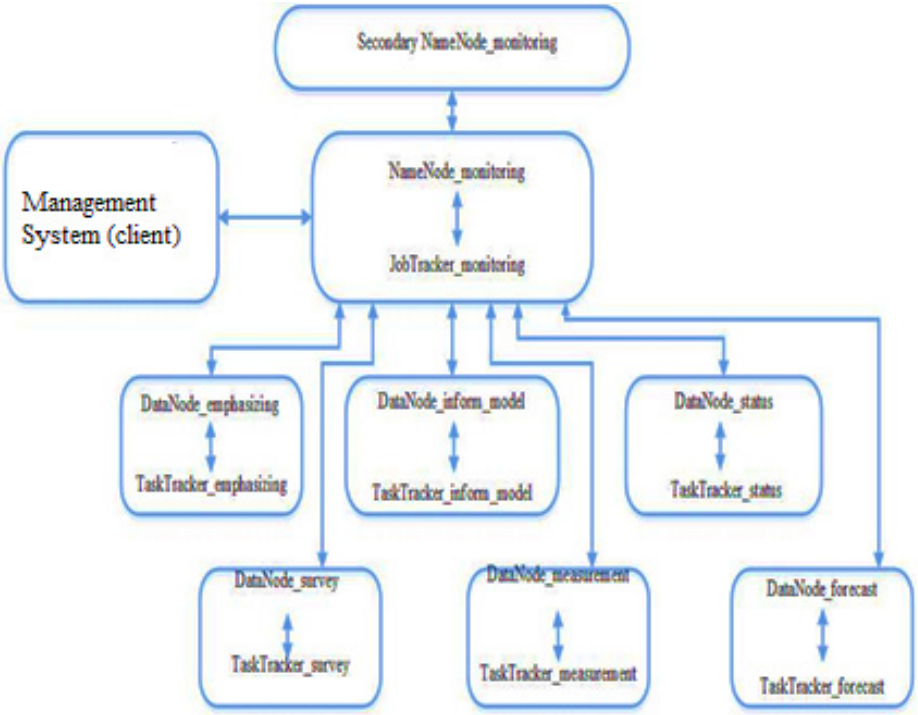

Fig. 3. Topology of the Hadoop monitoring cluster.
Daemons description is given in [17].

For the topology of the Hadoop monitoring cluster [16], the methodological recommendations were developed and studied for the synthesis of TaskTraker_state and DataNode_state daemons. The recommendations are responsible for solving the problem of assessing the state of the object of observation and identifying its information model taking into account the features of cloud computing. Principles and approaches were proposed [17] based on neuro-fuzzy solutions that can be used as a basis for the design of intelligent automated systems for monitoring objects in the web-space.

Decision-making mechanisms based on the formalization of a priori experience of experts in the fuzzy base of fuzzy production rules are proposed [17]. The possibilities of a neuro-fuzzy classifier in the form of a three-layer fuzzy neural network were investigated within the framework of solving the problems of classification and expansion of the classification of input data on the characteristics of the attributes of the monitoring object. The network consists of the following levels:

– system of fuzzy production rules describing the identifier operation taking into account expert assessments;

– neuro-fuzzy network, the structure of which reflects the system of fuzzy production rules;

– clear self-learning neural network for solving the problem of clustering (classification) of input data from the web-space.

Moreover, the lower level solves the problems of operational identification of attribute changes, and the upper one solves the problems of the accumulation of experience in detecting the effects of such changes on the elements and nodes of the monitoring object.

To train a neural network, a general approach is proposed that allows taking into account the stochastic dynamics of the attributes of the monitoring object in the web-space. The approach is based on the standard method of minimizing the generalization error on the basis of minimizing the quadratic residual functional on the training sample.

An approach to the mathematical formalization of the synthesis of the TaskTracker_state daemon in a task of conditional optimization was proposed [17]. Limitations in the form of inequalities reflecting the specifics of cloud computing in Hadoop environment are proposed. The possibilities of using the bootstrap method to assess the dynamics of the attributes of the monitoring object are analyzed.

The results obtained during the computer experimental study of the possibility of practical implementation of the neural network method for determining the type of computer attack showed a fairly high quality of the method's work [17].

Known approaches to parallel data processing in cloud systems [18] are reduced to the implementation of parallel operations, which is applicable to work with databases, tables, but does not reflect the specifics of the general problem of the synthesis of the Hadoop monitoring cluster, since the problem of scientific and methodological justification of the required cloud resource in parallel processing of streaming services with proper quality remains unresolved. To achieve this goal, it is necessary first to develop mathematical models for the bit rate of DStream formation and for a group of parallel DStreams from sources processed in a cloud cluster via the Hadoop technology.

Model for the bit rate of DStream formation

We assume that in the current interval of time $[t_0, t]$ as a result of measurements to a random process of bit rate of DStream formation from the s-th source of the k-th service with a variable bit rate, it is possible to match a finite set of discrete values that reflect the poly-burst nature of the bit rate.

The term "poly-burst" means that there are time intervals in which the source generates information with a high bit rate (burst), significantly exceeding the average bit rate for the entire time the service is provided to the consumer.

Let us denote a finite set of these values $\left\{B_{\max_j}^{(sk)}\right\}_{j=1}^{n_s(t)}$. Elements of this set correspond uniquely to elements of the set of time intervals $\left\{\left[t_{o_j}^{(sk)}, t_{p_j}^{(sk)}\right]\right\}_{j=1}^{n_s(t)}$, the set of probability values $\left\{p_j^{(sk)}\right\}_{j=1}^{n_s(t)}$ and the set of burst coefficients $\left\{k_{B_j}^{(sk)}\right\}_{j=1}^{n_s(t)}$ [19], where $k_{B_j}^{(sk)} = \dfrac{B_{\max_j}^{(sk)}}{B_{aver}^{(sk)}}$.

The result of a stepwise approximation of a random process of bit rate of DStream formation from the s-th source of the k-th service at time t will be written as

$$\tilde{b}_d^{(sk)}(t) = \sum_{i=1}^{n_s(t)} B_{\max_i}^{(sk)} \left[\Theta\left(t - t_{o_i}\right) - \Theta\left(t - t_{p_i}\right)\right]$$

and the probability distribution density of the random process of bit rate of DStream formation from the *s*-th source of the *k*-th service at time *t* can be expressed in terms of the sum of Delta functions [14]

$$f_t\left(\tilde{b}_d^{(sk)}\right)=\sum_{j=1}^{n_s(t)} p_j^{(sk)}\delta\left(\tilde{b}_d^{(sk)}(t)-B_{\max_j}^{(sk)}\right)\left[\Theta\left(t-t_{o_j}\right)-\Theta\left(t-t_{p_j}\right)\right], \quad (1)$$

Where $\Theta(t)=\begin{cases}0, t<0\\1, t\geq 0\end{cases}$ is the Heaviside step function, and

$\sum_{j=1}^{n_s(t)} p_j^{(sk)}k_{B_j}^{(sk)}=1, k_{B_j}^{(sk)}=\frac{B_{\max_j}^{(sk)}}{B_{aver}^{(sk)}}, \sum_{j=1}^{n_s(t)} p_j^{(sk)}k_{B_j}^{(sk)}=1$ is the characteristic property of poly-burst DStream traffic.

Bit rate probability distribution function from the *s*-th source of the *k*-th service at time *t* is obtained by integrating

$$F_{(t)}\left(\tilde{b}_d^{sk}\right)=\int_{-\infty}^{\tilde{b}_d^{(sk)}(t)} f_t(\tau)d\tau=$$

$$=\int_{-\infty}^{\tilde{b}_d^{(sk)}(t)}\sum_{i=1}^{n_s(t)} p_i^{(sk)}\delta\left(\tau-B_{\max_i}^{(sk)}\right)\left[\theta\left(t-t_{0_i}\right)-\theta\left(t-t_{p_i}\right)\right]d\tau=$$

$$=\sum_{i=1}^{n_s(t)} p_i^{(sk)}\left[\theta\left(t-t_{o_i}\right)-\theta\left(t-t_{p_i}\right)\right]\int_{-\infty}^{\tilde{b}_d^{(sk)}(t)}\delta\left(\tau-B_{\max_i}^{(sk)}\right)d\tau=$$

$$=\sum_{i=1}^{n_s(t)} p_i^{(sk)}\left[\theta\left(t-t_{o_i}\right)-\theta\left(t-t_{p_i}\right)\right]\int_{-\infty}^{\tilde{b}_d^{(sk)}(t)}\delta\left(\tau-B_{\max_i}^{(sk)}\right)d\tau=$$

$$=\sum_{i=1}^{n_s(t)} p_i^{(sk)}\left[\theta\left(t-t_{o_i}\right)-\theta\left(t-t_{p_i}\right)\right]\theta\left(\tilde{b}_d^{(sk)}(t)-B_{\max_i}^{(sk)}\right),$$

since the primitive of the Delta function is the Heaviside function.

The first moment and variance of stepwise approximation of a random process of the poly-burst bit rate of DStream formation from the *s*-th source of the *k*-th service on the time interval $[t_0, t]$ by [19] can be expressed as follows:

$$E\left[\tilde{b}_d^{(sk)}(t)\right]=\sum_{i=1}^{n_s(t)} p_i^{(sk)}B_{\max_i}^{(sk)}, \quad (2)$$

$$D\left[\tilde{b}_d^{(sk)}(t)\right]=\sum_{i=1}^{n_s(t)}\sum_{j=1}^{n_s(t)} p_i^{(sk)}p_j^{(sk)}\left(B_{\max_i}^{(sk)}-B_{\max_j}^{(sk)}\right). \quad (3)$$

By time *t* the average value and the variance of the rate of the random process of DStream formation from the *s*-th source of the *k*-th service can be easily converted to the average value and the variance of the rate of transmission of micro-batches for the selected time intervals:

$$E\left[r_{micro-batch}^{(sk)}(t)\right]=\frac{E\left[\tilde{b}_d^{(sk)}(t)\right]}{L_{\inf}},$$

$$D\left[r_{micro-batch}^{(sk)}\right]=D\frac{\left[\tilde{b}_d^{(sk)}(t)\right]}{L_{\inf}^2},$$

where L_{inf} is the bit length of the information part of the micro-batch.

Mathematical model of DStreams group

There is a technological possibility to combine many DStreams from multiple sources. Then, for the superposition $N_s(t)$ of independent streams at time t we obtain an expression for the probability distribution density of their group bit rate B

$f_t(B)=\Pi_{k=1}^{N_s(t)}f\left(\tilde{b}_d^{(sk)}\right)$, given the formula (1).

Accordingly, the probability distribution function of the poly-burst bit rate B of the superposition $N_s(t)$ of streams at time t is the following expression:

$$F_t(B)=\Pi_{k=1}^{N_s(t)}F_t\left(\tilde{b}_d^{(sk)}\right)=\Pi_{k=1}^{N_s(t)}\Sigma_{i=1}^{n_s(t)}p_i^{(sk)}\left[\theta\left(t-t_{o_i}\right)-\theta\left(t-t_{p_i}\right)\right]\theta\left(\tilde{b}_d^{(sk)}(t)-B_{\max_i}^{(sk)}\right).$$

In general, the number of DStreams in a significant time interval (t_0, t) in i-th cloud of an ecosystem of N clusters ($i = 1,\cdots, N$) from the sources of the k-th service is a random process

$$N_{DStream_i}^{(k)}(t)=\gamma_{\Sigma_i}^{(k)}(t)(t-t_0),$$

where

$$\gamma_{\Sigma_i}^{(k)}(t)=\Sigma_{j=1}^{N_{source_i}^{(k)}(t)}\gamma_s^{(k)}(t), \tag{4}$$

$\gamma_s^{(k)}(t)=\frac{N_{S1}^{(k)}(t)+N_{S2}^{(k)}(t)}{t-t_0}$ is the value of the intensity of the stream of service requests from the s-th source $1\le s\le N_{source_i}^{(k)}(t)$ of the k-th service,

$N_{S1}^{(k)}(t)$ is the number of DStreams serviced at time t, and

$N_{S2}^{(k)}(t)$ is the number of DStreams claimed, but maintenance-free at time t.

The value of a random summation process of DStreams from all K services in i-th cloud cluster at time t, taking into account the formula (4), is

$$\gamma_{\Sigma_i}(t) = \sum_{k=1}^{K} \gamma_{\Sigma_i}^{(k)}(t).$$

The total number of requests for processing DStreams from all the sources of all K services of i-th cluster at time t can be considered a random value – the value of a random process at time t

$$N_{\Sigma_{source_i}}^{(k)}(t) = \gamma_{\Sigma_i}^{(k)}(t)(t - t_0).$$

Taking into account (2) and (3), the numerical characteristics of the bit rate of DStreams processing that is required by the sources of the k-th service of the i-th cluster at a time t can be found as the numerical characteristics of the sum of the random number of independent random processes [20]:

$$E\left[b_{\Sigma_i}^{(k)}(t)\right] = \Sigma_{j=1}^{N_{source_i}^{(k)}(t)} E\left[\tilde{b}_d^{(jk)}(t)\right],$$

$$\sigma^2\left[b_{\Sigma_i}^{(k)}(t)\right] = D\left[b_{\Sigma_i}^{(k)}(t)\right] = \Sigma_{j=1}^{N_{source_i}^{(k)}(t)} D\left[\tilde{b}_d^{(jk)}(t)\right].$$

Since the cluster will serve a sufficiently large number of DStreams of each service (k = 1, …, K), the law of distribution of the sum of transfer rates can be approximated by the normal distribution law even if the source transfer rate is subject to any distribution law [20]. The main limitation imposed on summable quantities is that they should be more or less the same, which is naturally the case for sources of the same service.

It is shown [20] that in this case, at the moment of time t, the probability density of the random process of the bit rate of DStreams, which is required by the sources of the k-th service of the i-th cluster, has the form:

$$f\left(b_{\Sigma_i}^{(k)}(k)\right) = \frac{1}{\sigma\left[b_{\Sigma_i}^{(k)}(t)\right]\sqrt{2\pi}} \exp\left[-\frac{\left(b_{\Sigma_i}^{(k)}(t) - E\left[b_{\Sigma_i}^{(k)}(t)\right]\right)^2}{2\sigma^2\left[b_{\Sigma_i}^{(k)}(t)\right]}\right],$$

where

$$E\left[b_{\Sigma_i}(t)\right] = \Sigma_{k=1}^{K} E\left[b_{\Sigma_i}^{(k)}(t)\right], \sigma^2\left[b_{\Sigma_i}(t)\right] = \Sigma_{k=1}^{K} \sigma^2\left[b_{\Sigma_i}^{(k)}(t)\right].$$

Omitting the intermediate integro-differential conversion and the introduction of new variables, using the office of the special functions, we express the probability distribution

function of a random process $F\left(b_{\Sigma_i}(t)\right)$ with parameters $E\left[b_{\Sigma_i}(t)\right]$ and $\sigma^2\left[b_{\Sigma_i}(t)\right]$ using the normal distribution function $\Phi(x)$) [21]:

$$F\left(b_{\Sigma_i}(t)\right)=\Phi\left(\frac{b_{\Sigma_i}(t)-E\left[b_{\Sigma_i}(t)\right]}{\sigma\left[b_{\Sigma_i}(t)\right]}\right),$$

This makes it possible to find the probability of the event that at time t the value of a random process of bit rate of DStreams processing $b_{\Sigma_i}(t)$, which is required to meet the current needs of the sources of the i-th cluster, can be given by the i-th cluster, respectively having performance or bandwidth B_i

$$P\left(b_{\Sigma_i}(t)\le B_i\right)=\Phi\left(\frac{b_{\Sigma_i}(t)-E\left[b_{\Sigma_i}(t)\right]}{\sigma\left[b_{\Sigma_i}(t)\right]}\right)=\Phi(u).$$

However, it is also necessary to solve the inverse problem, i.e., to determine the probability of the event that at time t the value of a random process of the rate of DStreams processing $b_{\Sigma_i}(t)$, which is necessary to meet the needs of all K services of i-th cluster, will exceed its performance

$$P\left(b_{\Sigma_i}(t)>B_i\right)=1-\Phi(u). \quad (5)$$

Expression (5) means that with probability $1-\Phi(u)$ some source at time t will not receive from the cluster the computing resource necessary for processing the information stream. The resource can be expressed in the number of cores.

Conclusion

We developed a mathematical model of parallel DStreams from sources processed in a cloud cluster via the Hadoop technology using the micro-batch architecture of the Spark Streaming module. The model takes into account, on the one hand, the stream of maintenance requests from the sources of various services and, on the other hand, the needs of services in transfer rate, taking into account poly-burst bit rate traffic sources with different services.

The formulas of the probability distribution functions of a random process of bit rate of the DStream formation from the s-th source of the k-th service with variable bit rate and a group of streams.

Analytical relations are obtained to calculate the required performance of the Hadoop cluster at a given value of the probability of micro-batch loss.

On the basis of the developed models it is possible to develop new guidelines for the automatic selection (adaptation) of the time batch interval with a focus on a particular service selected in the range from 500 milliseconds to several seconds. Each batch will create a set of RDD and will be processed by Spark in terms that reflect the technological features of real services of the Hadoop monitoring cluster, which appear regardless of the cluster functioning.

The developed models allow us to carry out the numerical analysis of parallel stream processing in a cloud cluster at the pre-project stage and to develop requirements, recommendations and algorithmic framework for the refinement of the functional of the Hadoop monitoring cluster in order to organize a predictable parallel process for maintenance of the monitoring objects of different nature in the web-space.

References:

1. Gudkova I., Maslovskaya N. Probability model for analysing impact of delays due to monitoring on mean service time in cloud computing. *T-comm.* 2014;8(6):13-15 (in Russ.).

2. Basharin G., Gaidamaka Y., Samoylov K. Mathematical theory of teletraffic and its application to the analysis of multiservice communication of next generation networks. *Automatic Control and Computer Sciences.* 2013;47(2):62-69. https://doi.org/10.3103/S0146411613020028

3. Buyya R., Broberg J., Goscinski A. Cloud Computing. Principles and Paradigms. New Jersey: John Wiley & Sons, Inc., 2011. 637 p. https://doi-org/10.1002/9780470940105

4. Focus Group on Cloud Computing. Technical Report. Part 1: Introduction to the cloud ecosystem: definitions, taxonomies, use cases and higher-level requirements. ver. 1.0 (02/2012). International Telecommunication Union, 2012. 62 p. http://handle.itu.int/11.1002/pub/808604ae-en

5. Khaled S., Boutaba R. Estimating service response time for elastic cloud applications. In: Proceed. of the 1th International Conference on Cloud Networking CLOUDNET, IEEE, 2012; pp. 12-16. https://doi.org/10.1109/CloudNet.2012.6483647

6. Recommendation ITU-T, 2013, Cloud Computing framework and high-level requirements, Y.3501, p. 27.

7. Recommendation ITU-T, 2013, Cloud Computing infrastructure requirements, Y.3510, p. 22.

8. Recommendation ITU-T, 2015, Cloud Computing framework for end-to-end resource management, Y.3520, (09/15).

9. Recommendation ITU-T, 2016, Overview of end-to-end cloud computing management, Y.3521/M.3070 (03/16).

10. Recommendation ITU-T, 2016, End-to-end cloud service lifecycle management requirements, Y.3522 (09/16).

11. Recommendation ITU-T, 2018, Cloud computing – Overview of inter-cloud trust management, Y.3517 (12/18).

12. Recommendation ITU-T, 2018, Cloud computing – functional requirements of inter-cloud data management, Y.3518 (12/18).

13. Karau H., Konwinski A., Wendell P., Zaharia M. Learning Spark: Lighting-fast data analysis. US: O'Reilly, 2015. 257 p.

14. Erokhin S.D. A review of scientific research on artificial intelligence. In: Proceed. 2019 Systems of signals generating and processing in the field of on board communications. IEEE, 2019. 4 p. INSPEC Accession Number: 18638425. https://doi.org/10.1109/SOSG.2019.8706723

15. Chesnokov A.S., Gorodnichev M.G., Gavrish K.A., Zhidkova M.A. Intelligent vehicle condition monitoring system. In: Proceed. 2019 Systems of signals generating and processing in the field of on board communications. IEEE, 2019. 4 p. INSPEC Accession Number: 18638469. https://doi.org/10.1109/SOSG.2019.8706727

16. Lam C.P. Hadoop in Action. Publisher: Manning Publications Company, 2011. 334 p.

17. Nazarov A., Nazarov M., Pantiuhin D, Sychev A., Pokrova S. Automation of monitoring processes in web-based neuro-fuzzy formalism. *T-comm.* 2015;9(8):26-33 (in Russ.).

18. Munerman V.I. The implementation of parallel data processing in cloud systems. *Sovremennye informacionnye tehnologii i IT-obrazovanie* = Modern Information Technology and IT-education. 2017;13(2):57-63 (in Russ.). https://doi.org/10.25559/SITITO.2017.2.223

19. Nazarov A.N. Model of parallel processing of tasks in the cloud cluster Hadoop. In: Proceedings of the XIII International Industry Scientific and Technical Conference "Technologies of the information society" (March 20-21, 2019, Moscow, MTUCI). In 2 v. V. 2. M.: Publishing House Media Publisher, 2019; pp. 69-71 (in Russ.).

20. Grigoriev V.R., Nazarov A.N. Methodological aspects of parallel problem solving in the cloud cluster of cyber-attacks monitoring. In: Proceedings of the XVIII Scientific and Practical Conference "Information technologies in public administration. Digital transformation into human capital" (April 25, 2019). M.: Research Institute "Voskhod", 2019. 4 p. (in Russ.).

21. Sokolov G.A., Gladkih I.M. Mathematical statistics. M.: Ekzamen Publ., 2004. 432 p. (in Russ.).

Литература:

1. Гудкова И.А., Масловская Н.Д. Вероятностная модель для анализа задержки доступа к инфраструктуре облачных вычислений с системой мониторинга // Т-сомм: Телекоммуникации и транспорт. 2014. Т. 8. № 6. С. 13–15.

2. Башарин Г.П., Гайдамака Ю.В., Самуйлов К.Е. Математическая теория телетрафика и ее приложения к анализу мультисервисных сетей связи следующих поколений // Автоматика и вычислительная техника. 2013. Т. 47. № 2. С. 11–21.

3. Buyya R., Broberg J., Goscinski A. Cloud Computing. Principles and Paradigms. New Jersey: John Wiley & Sons, Inc., 2011. 637 p. https://doi.org/10.1002/9780470940105

4. Focus Group on Cloud Computing. Technical Report. Part 1: Introduction to the cloud ecosystem: definitions, taxonomies, use cases and higher-level requirements. ver. 1.0 (02/2012). International Telecommunication Union, 2012. 62 p. http://handle.itu.int/11.1002/pub/808604ae-en

5. Khaled S., Boutaba R. Estimating service response time for elastic cloud applications. Proceed. of the 1th International Conference on Cloud Networking CLOUDNET, IEEE, 2012; pp. 12-16. https://doi.org/10.1109/CloudNet.2012.6483647

6. Recommendation ITU-T. Y.3501. Cloud Computing framework and high-level requirements. Geneva: International Telecommunication Union, 2013. 27 p.

7. Recommendation ITU-T. Y.3510. Cloud Computing infrastructure requirements. Geneva: International Telecommunication Union, 2016. 28 p.

8. Recommendation ITU-T. Y.3520 (09/15). Cloud Computing framework for end-to-end resource management. Geneva: International Telecommunication Union, 2015. 26 p.

9. Recommendation ITU-T. Y.3521/M.3070 (03/16). Overview of end-to-end cloud computing management. Geneva: International Telecommunication Union, 2016. 32 p.

10. Recommendation ITU-T. Y.3522 (09/16). End-to-end cloud service lifecycle management requirements. Geneva: International Telecommunication Union, 2016. 24 p.

11. Recommendation ITU-T. Y.3517 (12/18). Cloud computing – Overview of inter-cloud trust management. Geneva: International Telecommunication Union, 2018. 24 p.

12. Recommendation ITU-T. Y.3518 (12/18). Cloud computing – functional requirements of inter-cloud data management. Geneva: International Telecommunication Union, 2018. 26 p.

13. Карау Х., Ковински Э., Венделл П., Захария М. Изучаем Spark: молниеносный анализ данных. М.: ДМК Пресс, 2015. 304 с.

14. Erokhin S.D. A review of scientific research on artificial intelligence. In: Proceed. 2019 Systems of signals generating and processing in the field of on board communications. IEEE, 2019. 4 p. INSPEC Accession Number: 18638425. https://doi.org/10.1109/SOSG.2019.8706723

15. Chesnokov A.S., Gorodnichev M.G., Gavrish K.A., Zhidkova M.A. Intelligent vehicle condition monitoring system. In Proceed. 2019 Systems of signals generating and processing in the field of on board communications. IEEE, 2019. 4 p. INSPEC Accession Number: 18638469. https://doi.org/10.1109/SOSG.2019.8706727

16. Лэм Чак. Hadoop в действии. М.: ДМК Пресс, 2012. 424 с.

[Lam C.P. Hadoop in Action. Publisher: Manning Publications Company, 2011. 334 p.]

17. Назаров А.Н. Назаров М.А., Пантюхин Д.В., Покрова С.В., Сычев А.К. Автоматизация процедур мониторинга в web-пространстве на основе нейро-нечеткого формализма // T-comm. 2015. Т. 9. № 8. С. 26–33.

18. Мунерман В.И. Реализация параллельной обработки данных в облачных системах // Современные информационные технологии и ИТ-образование. 2017. Т. 13. № 2. С. 57–63. https://doi.org/10.25559/SITITO.2017.2.223

19. Назаров А.Н. Модель параллельной обработки задач в облачном кластере Hadoop // В кн.: Сборник трудов XIII Международной отраслевой научно-технической конференции «Технологии информационного общества» (20-21 марта 2019 г. Москва, МТУСИ). В 2-х т. Том 2. М.: ООО ИД Медиа Паблишер, 2019. С. 69–71.

20. Григорьев В.Р., Назаров А.Н. Методические аспекты параллельного решения задач в облачном кластере мониторинга кибер-атак // Сборник трудов XVIII научно-практической конференции «Информационные технологии в государственном управлении. Цифровая трансформация в человеческий капитал» (25 апреля 2019г.). М.: ФГУП НИИ «Восход», 4 с.

21. Соколов Г.А., Гладких И.М. Математическая статистика: Учебник для вузов. М: Издательство «Экзамен», 2004. 432 с.

About the author:

Alexey N. Nazarov, Dr. of Sci. (Engineering), Professor, Professor of the Chair “Information warfare”, Institute for Integrated Security and Special Instrumentation, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow 119454, Russia). E-mail: a.nazarov06@bk.ru

Об авторе:

Назаров Алексей Николаевич, доктор технических наук, профессор, профессор кафедры «Информационное противоборство» Института комплексной безопасности и специального приборостроения ФГБОУ ВО «МИРЭА – Российский технологический университет» (119454, Россия, Москва, пр-т Вернадского, д. 78). E-mail: a.nazarov06@bk.ru