International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I12P209

Volume 70 Issue 12, 76-89, December 2022

Original Article

Development of Components of a Distributed Fault-Tolerant Medical Data Storage System

Aslan A. Tatarkanov[1], Abas Kh. Lampezhev[2], Dmitry V. Polezhaev[3], Ruslan Kh. Tekeev[4]

[1,2,3,4]*Institute of Design and Technology Informatics, Russian Academy of Sciences, Moscow, Russia*

[1]*Corresponding Author : as.tatarkanov@yandex.ru*

Received: 03 September 2022 Revised: 09 December 2022 Accepted: 16 December 2022 Published: 24 December 2022

Abstract *- In human activities, various technologies based on a distributed approach are increasingly mastered; they are designed to ensure efficient and reliable information storage, quick data access, and the possibility of implementing parallelism when working with them. Comprehensive research aimed at optimizing the tactical, technical, and economic indicators determining the appearance and functionality of such storage systems is an urgent scientific problem. One of its parts is the need to investigate the possibilities of developing components of an effective fault-tolerant medical data storage system. This determined the article's topic relevance. This research aimed to develop new mathematical models and algorithms, based on known alternative technologies, for the basic components of a distributed storage system that can be effectively implemented. The application of these components will increase system fault tolerance using controlled redundancy. The article shows that, among the possible options for a distributed fault-tolerant data storage system structure, distributed data storage systems are most promising in ensuring efficient and reliable information storage. Moreover, this refers to such systems whose mechanism of maintaining reliability (fault tolerance) is based on the operation of the error-correcting code based on a redundant residue number system. A model of a distributed data storage system with a redundancy function is proposed. It is substantiated that the latter's functioning depends on the efficiency and practical applicability of the approaches to converting values from a positional system to a redundant one.*

Keywords *- Algebraic codes, Data errors, Data redundancy, Data storage system, Non-positional notation.*

1. Introduction

A stable significant increase in the volumes of received, processed, and, consequently, the stored information is a characteristic feature of the modern world [1,2]. The effectiveness of the analysis of large amounts of data for various purposes is characterized by such indicators as the speed of analysis, its quality, the reliability of the information provided, etc. [3,4].

Therefore, recently, brisk growth has been observed in developments related to new methods for information transmitting, processing and storing, such as technologies for working with large amounts of information [5], cloud and fog computing [6,7]; technologies for Web information processing [8]; neural network-based methods [9]; technologies for ensuring the reliability and security of information and data storage [10], their availability, etc.

While computing capabilities continue to grow, increasing attention is being paid worldwide to finding such design solutions that can provide greater fault tolerance and increased performance of data storage systems (DSS) [11]. The main DSS components of storage systems, which are hardware and software complexes, include information storage devices, storage device access systems, backup systems, storage management software, monitoring and control systems, etc.

The need to improve such structurally complicated DSS is an urgent scientific problem. The effectiveness of solving it is associated with the need to research to develop approaches to ensure data integrity, increased fault tolerance, scalability, and performance of such systems. Secondly, it is necessary to search for such design solutions for a technically and economically optimal DSS that will significantly reduce energy costs, service costs, the cost of backup storage systems, and much more.

Hardware, software and network failures and interruptions in the DSS power supply led to such negative consequences as a violation of the data reliability, temporary or permanent termination of access to them, etc. [12]. To increase fault tolerance and avoid such problems, it has become increasingly obvious to apply a distributed approach to data storage using computer networks in which information can be stored on more than one node [13]. It provides fast access to data and the ability to implement parallelism, breaking a complex task into parts to ensure its solution using multiple processors [14]. A good potential for parallelization lies in neural networks [15,16].

Thus, currently, the key task is to develop technologies for shaping the appearance of such distributed data storage systems (DDSSs), which allow for the most efficient operation of available local resources. The DDSS is based on various models and algorithms for improving fault tolerance and performance. The DDSS basis is the subject of this research.

The research aimed to develop components of a DDSS whose application would increase its fault tolerance using adjustable redundancy.

2. Review and Analysis of Possible Alternative Options for the Structure of a Distributed Fault-Tolerant Data Storage System

DDSSs are the most promising systems in terms of providing efficient and reliable storage of information arrays of various types, structures and sizes [17,18], which include multiple storage devices connected by a network and infrastructure ensuring their performance.

The method presented for data storage enables the storage of different parts of information on different media. Rack, server, or other storage is irrelevant. Simultaneously, it should be noted that from the standpoint of a specific user, a network of equipment intended for data storage is a conditionally single device [19]. Access to information does not depend on the location of request formation or the method for organizing the placement of certain information.

The key components of any DDSS designed to ensure data storage reliability, that is, to preserve its integrity and validity to the utmost, are listed below:

- A set of algorithms and methods for processing input information, its presentation in any form, for example, in a redundant one. Moreover, in any case, it should be suitable for storing data in conditions of distribution and placement on different devices and media;
- A storage subsystem. It includes a set of certain media associated with a single system. Options include file, logical and other subsystems. The system must determine the method of storage and "observes" the provided access policy;
- A set of algorithms and methods for data recovery that provide reverse recovery of distributed information within the framework of such procedures as:
    - Collection of distributed data components, using metadata about the places of their storage and conditions of access to them;
    - Assessment of damage to the collected data;
    - Data recovery based on the original information and identified damaged blocks.

The DDSS structure is shown in Fig. 1 in its most general form.

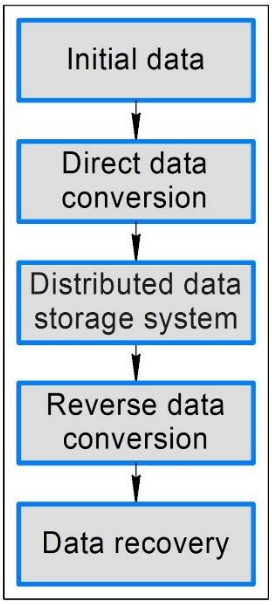

Fig. 1 Schematic diagram of the DDSS

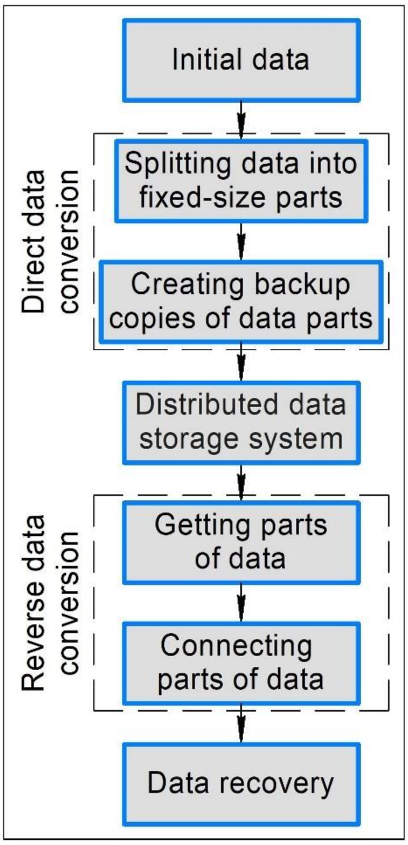

Fig. 2 A replication-based DDSS workflow

The algorithms and methods listed above can be implemented differently depending on the mechanisms used to maintain reliability (fail tolerance). A critical method to ensure reliability (the application of new, more reliable components; the selection of suitable operating modes; the development of entirely new types of architecture that are compatible with the use of modern technologies, etc.) is to create backups, that is, maintaining DDSS fault tolerance by using supporting tools and resources, including "cold" [20] and "hot" [21] backups. At the same time, it should be noted that any way to create DDSS backup copies inevitably leads to redundancy [22].

Among the practical approaches that cause redundancy, the most widely used methods are data replication (creation of backup copies) [23], data separation algorithms [24], and error correcting codes [25], which are primarily caused by bit damage during information transmission, erasure codes, and others.

The features of each method directly affect the cost and the level of complexity of the system being developed. The simplest way to create backup data is replication based on their copying and subsequent storing in different storage locations [26]. At the same time, full accessibility and integrity of the stored information are guaranteed. If access to the requested data is unavailable on one physical medium, it will be provided using another. Fig. 2 shows the replication-based DDSS workflow.

A similar method for creating backup data is associated with increased DDSS redundancy. Moreover, despite its extreme simplicity, storage costs almost double (compared to the situation of a complete refusal of data backup).

Methods for creating backup data based on data partitioning algorithms are significantly more complex than replication. This can be exemplified with the Rabin algorithm [27], which involves data conversion using the vector-matrix product in Galois fields.

Also, applying a wide range of error-correcting codes (Hamming codes, Reed-Solomon codes, etc.) is associated with the expenditure of significant computing resources, for which the analyzed data redundancy is a necessary condition for the possibility of their use [25,28].

Erasure codes are also characterized by increased complexity, which can be considered a complex tool for creating data redundancy, within which additional data is generated and stored in addition to the input information, for example, about parity, which, in the absence of access to the requested information, makes it possible to restore it [29].

However, simultaneously, owing to the existing methods for simplifying DDSS implementation, a DDSS, which is based on this method for maintaining increased fault tolerance, is one of the most effective, both from technical and economic viewpoints.

Since information systems become more productive with each new generation, the requirements for DDSS and its components are constantly increasing. Often, cutting-edge software is required to ensure data storage reliability, process data in a distributed form, and safeguard confidential information [30].

Combined with anti-noise coding, all these circumstances can make the system less productive. The use of multifunctional error-correcting codes, compatible, in particular, with arithmetic coding and the ability to ensure the safety and security of stored information from leaks, is the most promising solution to this problem [31].

These are provided by codes based on the residue number system (RNS), and the redundant residue number system (RRNS) possesses such capabilities [32,33]. These number systems have found application as a non-positional notation and are intended to organize computational operations with data in a distributed form and make these operations more accurate. At the same time, RNS (RRNS) are the foundation for schemes that provide increased protection of information confidentiality in distributed remote storage cases.

The results of several research papers in this research domain are summarized in Table 1.

Table 1. Results of publications in the considered research domains

Reference	Paper Content
[8]	The research is devoted to solving the problems of connecting numerous devices to the Internet in the process of implementing the information-oriented Internet of Things project. This research aimed to conduct comprehensive studies of a conceptual nature in developing technologies to ensure the efficient operation of numerous Internet of Things project applications. The authors identified some of the most promising research areas to overcome the problems of inefficient, traditional caching solutions for this project.
[6]	The research is devoted to solving the problems of optimizing cloud, fog and edge computing and implementing the Internet of Things project. The research purpose was to conduct comprehensive conceptual studies to develop a clear understanding of the performance metrics and standards that can be used in this area of optimization. The authors comprehensively considered various performance indicators and metrics for cloud, fog, and edge computing that may be useful in the future. Some of the most promising areas of research were identified.
[29]	This research solves a set of issues related to analyzing the technical and economic possibilities of cross-centre data storage technology. This research aimed to study the effectiveness of building this technology using erasure codes. The authors proposed an adaptive to the network environment method for erasure-coded cross-datacenter storage and a hybrid-structured method for erasure code recovery in cross-datacenters. The results of experimental studies showed that the proposed methods could reduce the cost and time of recovering damaged data by 40% or more compared to several advanced erasure codes.

[15]	The research is devoted to solving the issues of increasing the efficiency of using general regression neural networks in solving predictive problems. This research aimed to improve forecasting accuracy when using neural networks. The authors proposed a parallel integrated neural network system, the model of which is a combination of a general regression neural network and an adaptive gray wolf dynamic optimizer. The experimental studies showed that the proposed system provided a higher average prediction accuracy than other methods (for example, by 8.05% for a wavelet neural network).
[32]	The research is devoted to addressing issues of increasing the efficiency of the applied use of convolutional neural networks. This research aimed to reduce the resources required for their creation and operation. The authors proposed such a network architecture based on the residual number system and the new Chinese theorem on the remainder with fractions. It was estimated that using such a hardware-software architecture could reduce hardware costs by 32.6% compared to the traditional approach. The average image recognition time was decreased by 37.06% compared to software implementation.
[1]	The research is devoted to a comprehensive analysis of the increasing impact of technologies, defined by the term "Big Data", on biomedical studies and their application in healthcare. This research aimed to analyze advances in the field of ohmic technologies (genomics, epigenomics, transcriptomics, proteomics, metabolomics and pharmacogenomics) and their contribution to the development of methods based on the concept of "Big Data" in biomedicine. The authors critically reviewed the main computational methods, algorithms and their results that have contributed to recent advances in the field of big data obtained as a result of biomedical research on various complex human diseases. Some of the most promising areas for further research were identified.
[21]	The research is devoted to improving distributed storage systems based on traditional technologies that use the redundant backup to reduce the cost of reliable storage and increase the utilization of space. This research aimed to develop a reliable data backup method that provides a high space utilization ratio and is suitable for efficiently storing hot temporary data. The authors proposed a method for data backup based on the use of a checksum, which ensures data security and saves storage space, thereby increasing read and write performance on average by 10% and 30%, respectively.
[17]	This research is devoted to analyzing the possibilities of using energy-saving technologies to ensure efficient data accessibility within all types of cloud services and the high quality of their servicing from environmental and economic viewpoints. This research aimed to develop a model and algorithm for energy saving for a distributed storage system. Experimental studies showed that the proposed algorithm improved the energy efficiency of a distributed storage system.
[9]	This research is devoted to a comprehensive study of problems related to interpreting deep neural network models, using the recently proposed technology of layered relevance distribution, creating transparent machine learning systems, etc. This research aimed to analyze and summarize the materials on these problems outlined in the textbook read by the authors at the 42nd IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017).
[28]	This research is devoted to studying problems related to constructing a series of twisted linearized codes. The research aimed to develop an approach based on using an algebraic description for sum-rank metric codes as quotient space of a skew polynomial ring. The proposed approach simultaneously generalizes the skew group algebra setting for rank-metric codes and the polynomial setting for codes in the Hamming metric.
[23]	The research is devoted to solving the problems of increasing the efficiency in terms of high throughput of the storage process in distributed erasure coding storage systems. This research aimed to develop proposals for implementing an "in-network" redundancy generation process, which distributes the data insertion load among the source and storage nodes. This approach allows storage nodes to generate new redundant data by exchanging partial information among themselves, improving the throughput of the storage process (up to 90% in data centers and 60% in peer-to-peer settings) compared to the classic data insertion approach.
[31]	This research aimed to experimentally study the performance of alternative fault-tolerant buffer schemes for organizing data in a Network-on-Chip (NoC) communication system protected by alternative types of error correction code. The results obtained showed that the use of error correction code in NoC buffers can be an effective solution for reliability issues, although increasing the design costs and requiring a buffer with higher storage capacity.
[11]	This research is devoted to a comprehensive analysis of fault tolerance of fast-changing big data systems as a key feature of such systems to ensure availability, reliability, and consistent performance during faults. This research aimed to provide a consistent understanding of fault tolerance in large data systems and highlight the problems that hinder improving its efficiency. The authors identified several of the most promising research areas for solving fault tolerance problems.

[2]	The research is devoted to a comprehensive analysis of the problems in terms of scalability, rapid ingestion, performance, processing and storage efficiency of new technologies, characterized by the term "Big Data". This research aimed to describe two future solutions for big data storage (REHDFS system and DNA Storage) with an assessment of their technical and economic advantages.
[33]	This research aimed to develop proposals regarding a novel application of RRNS codes to Space-Time Block Codes (STBCs) design. According to the proposed method, the link between residues and complex signal constellation was optimized based on the so-called "Direct Mapping" and "Indirect Mapping" schemes. The knowledge of the apriori probabilities of residues was used to implement a probability-based Distance-Aware Direct Mapping scheme. Simulation results demonstrated that the Distance-Aware Direct Mapping scheme provided performance gain relative to a traditional direct mapping scheme.
[27]	The research is devoted to improving the graphics processing units to increase their power, as the most important indicator of processors when performing parallel procedures. This research aimed to develop a technology that combines the recent advancements in graphics processing units with string-matching algorithms that significantly increase the efficiency of any application. The article's authors proposed a modified parallel version of the Rabin-Karp algorithm using a graphics processing unit to solve this problem.
[5]	The research is devoted to a comprehensive analysis of technologies defined by the term "Big Data" in terms of uploading big volumes of data, processing and analyzing unstructured information and its distribution into the clustered database, etc. This research aimed to organize a parallel system that includes several methods for working with arrays of unstructured data. Within the framework of the proposed system, a class of basic database operations was implemented: database connection, table creation, obtaining a row identifier, returning all database elements, updating, deleting, and creating a row.
[12]	This work conducted a set of studies to develop a formal description of methodological approaches to form the image of automated diagnostics of medical and biological systems subjected to random perturbation. This study reviewed current diagnostic methods of the main diagnostic system elements. Research on the development of statistical recognition systems, providing a link of the detection reliability with the necessary constraints to achieve this, is relevant. The study showed that the formation of features using a nonlinear transformation procedure in initial signal spaces and a stochastic coding method of classification of the features is based on calculating the correlation moment using the correlation functions of signs.
[25]	The research is devoted to improving a ternary optical computer, particularly to increasing the stability of the communication channel. This research aimed to increase the strength of the communication channel of such a computer to random failures by developing and using a Hamming code error correction module in the system. The experimental results showed that the bit error rate of the output results of a ternary optical computer could be reduced by one order of magnitude after error correction by the Hamming code error correction module.
[16]	The research is devoted to the analysis and development of parallel algorithms that implement the Z-transform neural network calculation based on the input and output signals of the analyzed object. This research aimed to conduct theoretical and experimental studies of the possibilities of parallel algorithms, various types of architectures with different numbers of processors, the computation of the Z-transform neural network and ZTANN training. The authors created efficient parallel algorithms and developed plans for creating a cost-optimal hardware platform based on microcontrollers operating in parallel.
[24]	The research is devoted to the parameter identification of Hammerstein output error moving average systems with a two-segment piecewise nonlinearity. This research aimed to solve the problem of converting the Hammerstein model into two regression identification models. The authors developed a recursive least squares algorithm based on data filtering to estimate the parameters of these two identification models. The proposed algorithm achieves higher computational efficiency than the standard approach by using smaller covariance matrices from two identification models instead of one identification model in the standard approach.
[26]	The research is devoted to improving (increasing the reliability) distributed storage systems for big data based on replication technology. This research aimed to develop a new reliability model to investigate the system reliability of multi-way declustering data layouts and analyze their potential parallel recovery possibilities. The comprehensive simulation results showed that the shifted declustering data layout outperforms the random declustering layout in a multi-way replication scale-out architecture regarding data loss probability and system reliability by up to 63% and 85%, respectively.
[18]	The research is devoted to improving traditional replication technology, which is key in forming distributed storage systems. This research aimed to develop an indirect replication algorithm based on the intrinsic characteristics of distributed storage systems and the peer-to-peer model. Compared to the traditional replication algorithm, the developed indirect replication algorithm has less granular replication, less bandwidth and storage costs and provides higher availability, durability, and security.

[3]	The research is devoted to analyzing technical and economic aspects in terms of the possibilities of implementation in the medical subject area, which is a prominent example of "big data" – a promising sequencing technology. This research aimed to summarize the main methodological and practical challenges of using big data in sequencing technology. The authors concluded that using big genomic data to support and inform consumer electronics associations (CEAs) of next-generation sequencing (NGS) technologies held great promise.
[10]	The research is devoted to analyzing the integrity of data stored within cloud services as the primary indicator of ensuring their security. This research aimed to develop a secure fuzzy identity-based public verification scheme. The safety and usability of the proposed scheme were confirmed experimentally.
[20]	The research is devoted to solving the problem of creating a rational, cost-efficient image of the information storage systems that are part of the processing centers, most of which are cold and archival data. This research aimed to develop proposals for existing storage infrastructure in cloud service centers, providing sufficient bandwidth at an extremely low cost to meet cold and archival workloads. The main component of the proposed system is a novel fat tree interconnect fabric to connect hard disk drives (HDDs) to existing servers and network infrastructure.
[13]	The research is devoted to solving problems of increasing efficiency in terms of maintaining high system performance of the data placement process in erasure-coded distributed storage systems. This research aimed to increase the practicality and efficiency of inserting coded blocks into a set of redundant storage nodes. To eliminate the shortcomings of existing data allocation schemes, such as load imbalance and a large delay in data transmission, the authors proposed a tree-structured data placement scheme with top-down cluster transmission. The proposed scheme could effectively improve the survival life of storage networks and reduce the data insertion time. It performed better than the optimal tree-structured scheme based on other optimization algorithms.

Thus, a comprehensive analysis of alternative options of the applied mechanisms for maintaining *DDSS* reliability (fault tolerance) showed that the *DDSS*, which is based on the error-correcting code, underpinned by the RRNS, is the most promising structure of a distributed fault-tolerant data storage system.

3. Model of a Distributed Backed-Up Data Storage Method with Redundancy Based on an RRNS

If storage organizes by a backing up, a particular file is divided into fragments (of a single size). Copying is also assumed. The recording is carried out on permanent storage media. Usually, these are HDDs. The number of copies – the replication factor – is a settable parameter. It is defined by each local "array" or file. Data regarding the location of copies on HDDs shall be stored on the main server. In this context, it is the central node. Access to parts of a file, which, for objective reasons, requires restoration, is coordinated through this node.

In the DDSS, based on the principle of data partitioning (unlike the model of redundant distributed data storage), a certain number n of subparts that are recorded on different HDDs are formed rather than copies of file parts. Moreover, if the user has access to a certain number of k of these subparts, if the condition $k < n$ is satisfied, the user can restore some parts of the entire file.

It has already been noted the totality of information regarding the location of file fragments on HDDs, regardless of their status ("operating", "out of order"), is stored on the central node – the main server. It is responsible for coordinating access to individual fragments that, for objective reasons, require restoration. Here, the numbers n (total) and k (total fragments) may differ from each other. It is primarily associated with the existing requirements for a specific DDSS.

The fragment sizes in the presented situation are smaller than those for files. Moreover, as the backup is made, individual copies acquire dimensions similar to those of the originals. Due to this, conditional redundancy occurs, but the damage to reliability is excluded. In this case, operating costs are partially minimized, and resistance to probable failures is increased.

The data partitioning model is denoted as (k, n). The calculation systems for the formation of subparts may differ due to the peculiarities of the separation scheme used.

Thus, the peculiarities of RNS and RRNS are that they are not positional number systems (PNSs). In it, any value A is expressed through a set of k residues by dividing α_i of this value by the value p_i, which is part of the set of modules (residuals of the base degrees) into which it is divided:

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_k), \alpha_i = A \, mod \, p_i \, , i = 1,2, \ldots, k.$$

The set of bases $\{p_1, p_2,..., p_k\}$ defines a particular RNS. In accordance with the Chinese remainder theorem (CRT), such a representation for any value A from the working interval [0, P_k] of the representation of numbers in RNS (where $P_k = p_1 \cdot p_2 \cdot \ldots \cdot p_k$,) can be unique only in a situation where all p_t values are pairwise coprime. If redundant modules p_k+1, pk+2, ..., pn are added to the system of working bases $\{p_1, p_2,..., p_k\}$, and the value $A \in [0, P_k]$ expands by the remainders from dividing into new modules $\alpha_{k+1},\alpha_{k+2},...,\alpha_n$, this will result in an RRNS.

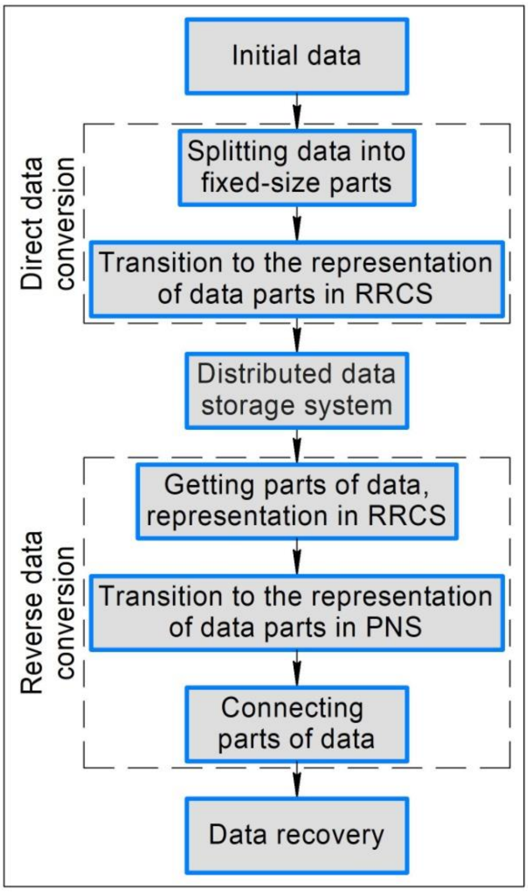

Fig. 3 A general model of the RNS (RRNS)-based DDSS

This type of number representation, which allows it to be divided into small parts of approximately the same size, is compatible with many applications, including those intended for DDSS, the RNS (RRNS)-based general model of which is shown in Fig. 3.

Use the following CRT formula to restore the positional form of a number based on its Robust Security Network (RSN) code:

$$A = \left|\sum_{i=1}^{k} P_i |P_i^{-1}|_{p_i} \alpha_i \right|_{p_k},$$

Where $P_i = \frac{P_k}{p_i}$.

It is worth noting that (as applicable to the RRNS) the bases in the system $p_1, p_2,.., p_n$ must be pairwise coprime. Additionally, it is necessary to include A in the interval [0, P_k] and exclude it from the interval [[P_k, P], where $P=Pk \cdot pk_{+1} \cdot pk_{+2} \cdot$ ... is a complete RRNS interval.

If the above conditions applied to the complete RRNS interval are met, the presence of distortion in the residuals of the A value presented in the RNS is detected. To this end, it is necessary to restore the value of A based on a complete system of RNS bases. In a situation where the final value A^* $\in [0, P_k]$, it should be recognized as correct, which means $A=A^*$. Otherwise, when $A^* \geq Pk$, this indicates the occurrence of distortion in one or immediately in several bases.

The performance of each RRNS-based structure is determined by how efficient the applied methods for converting values from the PNS to RRNS and inverse transformation, designed to restore the initial representation of the values, will be.

4. Development of an Algorithm for Selecting a Data Separation Scheme Structure, Concerning the Data Volume, Requirements for the Reliability Level, Including Reliability of HDDs of a Distributed Storage System

The level of the DDSS redundancy can be reflected in the redundancy rate, which is understood as the ratio of the number of redundant components available to the number of working features. This parameter plays an essential role in the DDSS design. In addition to maintaining fault tolerance at the highest possible level, the system being developed must be highly efficient from an economic viewpoint.

The indicator characterizing the redundancy rate, in general terms and terms of distributed systems with data partitioning models, can be represented by the following expressions respectively:

$$Re\,d\,undancy = I_{Full}/I_{Useful} - 1;$$

$$Re\,d\,undancy \approx -\big(1 - (n/k)\big) \cdot 100\%,$$

Where I_{Full} and I_{Useful} are full and useful data volumes, respectively; n is the total number of subparts that are formed for each specific part; k is the number of subparts that are sufficient to restore the entire part.

The sign "approximately" in the expression is used because, regarding a certain set of partitioning models, it is irrational to strive to fulfill the condition according to which the total bit depth of fragments k for restoring a part must be equal to the file bit length. RRNS can be considered the simplest example of an exception. Based on the requirement that all system modules should be characterized by maximum simplicity, the redundancy of some nodes should exceed the same for working nodes; it can be said that it is necessary to select the optimal coverage of the operating range in all aspects in each case. In practice, this problem has no solution.

RRNS modules will overlap the mentioned range excessively; however, a minimum excess of the optimal value can be achieved with a competent approach.

Thus, it is evident that the methodology for maintaining fault tolerance should ensure redundancy rate minimization and achieve an optimal level of reliability. In the case of distributed storage with increased fault tolerance, the main goal is a minimax multicriteria task, within which two groups of indicators define an objective function.

The first category includes such characteristics as the level of the system reliability and fault tolerance, the degree of data availability, the degree of data redundancy, the level of performance, etc.; the second category includes equipment costs, operating costs, energy consumption, etc.

The generalized limit criterion for the efficiency of a DDSS, which sets the key parameters for specifying the structure, can be represented as follows:

$$Z = \min_{y \in Y}(\max_{x \in X} F(x, y)); X = (x_1, x_2, ..., x_n), Y = (y_1, y_2, ..., y_m).$$

Variables x_1, x_2,..., x_n can reflect a number of parameters. These are, for example, performance, reliability, fault tolerance, and so on. Variables y_1, y_2,..., y_m are consumed resources regardless of the sequence number. It refers to power, time and so on.

In a conditionally general case, the structure of a distributed system with a partitioning function represents schematically, as shown in Fig. 4.

Here, variable *NP* is the number of parts obtained as a result of file partitioning; *n* is the total number of subparts that forms for each specific part; *k* is the number of subparts that are sufficient to restore the part.

To conduct analytical studies for choosing the optimal parameters of the RRNS structure shown in Fig. 4, it is proposed to use such an indicator as the probability of failure "on demand" to a request within a certain time, which can be defined as follows:

$$\boldsymbol{PFD = 1 - \left(\sum_{j=k}^{n}\left(C_n^j \cdot (1 - AFR)^j AFR^{n-j} \cdot \sum_{i=j\left\lfloor\frac{j-k}{2}\right\rfloor}^{j} C_j^i (1 - er)^i er^{j-i}\right)\right)^{CC}},$$

Where:
AFR is the average number of failures of one HDD per year;
er is a value showing the risk of distortion;

j – the number of available HDDs with records corresponding to specific fragments of the file being restored.

The formula derivation is based on a number of provisions:

- Direct access as a function in the context of storing files on hard drives does not necessarily imply recovery. The fact is that data can be damaged while the HDD is operating in a conditionally standard mode;

- Distortions can be affected by interference if they propagate through communication channels;

- File recovery is possible if all parts are in order;

- If the increase in stability is provided by a scheme of type (*k*, *n*), access to at least *k* HDDs is required to restore the fragment. The mentioned scheme guarantees the correctness of the restored data, provided that at least (*n* - *k*) / 2 fragments are not affected by distortion;

- It is important to fulfill the condition: the number of HDDs with corrupted data should not be greater than the rate of errors that can be corrected using undamaged fragments.

The method developed based on the formula presented above makes it possible to select partitioning schemes of type (*k*, *n*) for files regarding their size and requirements for the reliability level and the reliability of the DDSS HDD.

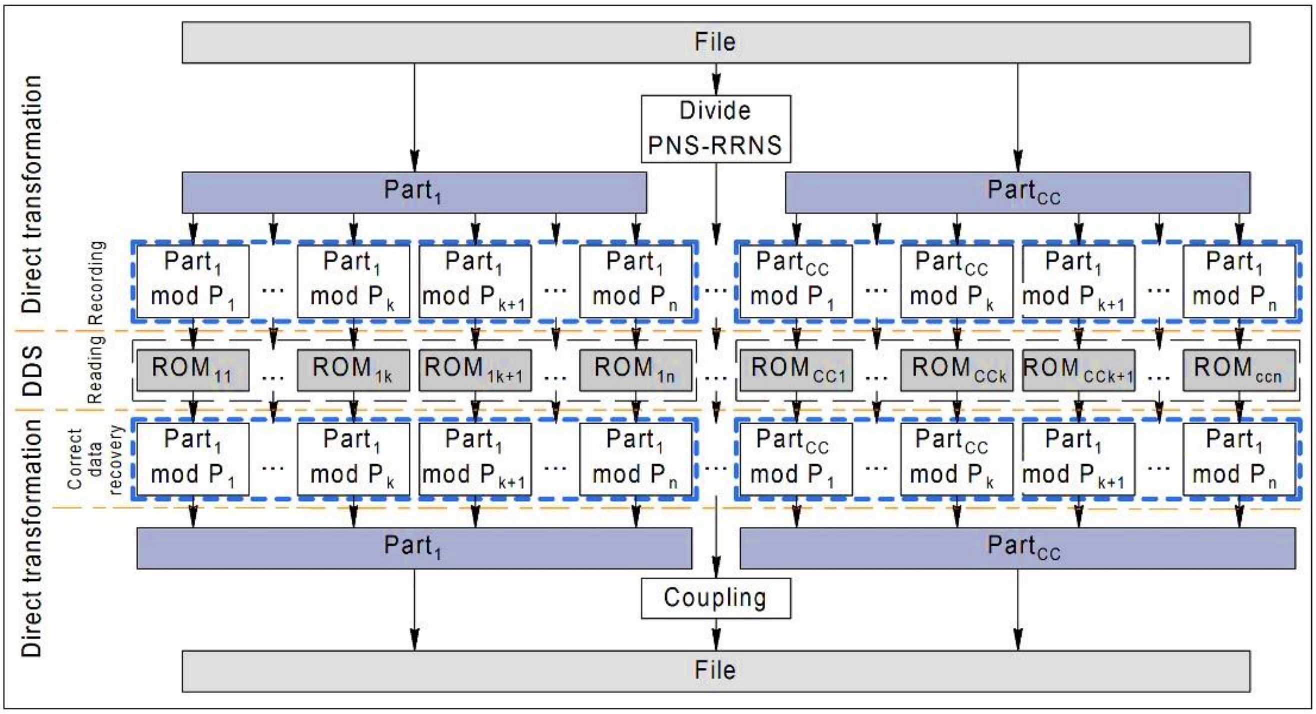

Fig. 4 General scheme of distributed storage using data partitioning (exemplified by RRNS)

5. Development of a Modified Direct Conversion Method for Organizing a Data Partitioning Subsystem

The process of converting DDSS data in direct form (PNS-RRNS) consists of two stages. In the first stage, all information is divided into parts of the set volume: *b* bit.

In the second stage, each part is automatically converted into the RRNS by calculating the residues after dividing each part into bases $p_1,..., p_k, p_{k+1},...,p_n$. In this case, a special partitioning method can be used. However, this procedure cannot always be performed by a machine method due to its complexity.

Separate methods for calculating residuals were proposed. The partitioning procedure was not applied. The method for summing the indicators of the positional number became the "basis". The method for summing up the values of the digit positions $A = \sum_{l=0}^{k} A_l N^l$; $0 \leq A_l \leq N-1$ turned out to be the most effective in terms of technology and economics. Its notation in the positional system with the base taken as *N* is as follows:

$$A = \left(\sum_{i=0}^{k} A_1^{(i)} B_1^{(i)} \, mod \, p_1, \sum_{i=0}^{k} A_2^{(i)} B_2^{(i)} \, mod \, p_2, \ldots, \sum_{i=0}^{k} A_n^{(i)} B_n^{(i)} \, mod \, p_n \right)$$

Where $N^i = \left(B_1^{(i)}, B_2^{(i)}, \ldots, B_n^{(i)}\right), A_i = \left(A_1^{(i)}, A_2^{(i)}, \ldots, A_n^{(i)}\right)$ are the degrees of the base N^i and the coefficients A_i in the system of residue numbers.

The peculiarity of the approach is that its "basis" involves the application of the principles of tabular mathematics [34]. As a result, costs are significantly reduced. An additional stage consists of the implementation of calculations in advance. Due to the presence of an array processor of n ($k + N + 1$), the conversion is carried out by the node with the highest quality and efficiency. However, using converters of this kind is associated with high costs.

This paper proposes a modified, more efficient direct summation method. The improvement of the original method, based on the results of analytical studies, showed that with an increase in the number of digit positions in each part into which the initial number partition, the number of modulo additions was reduced, which was achieved due to a slightly different way of the number representation.

A variant of the approach to summation in the form of a modification provides the efficiency of searching for residues from dividing numbers with many-digit positions. The method's effectiveness reveals itself when working with number systems if the base *N* is not large.

Thus, for example, in the case of a binary system, when $N = 2$, a bit binary number with the number of "fragments" *b* can be found as a combination of weighted values $\lceil b/s \rceil$characterized by the dimension *s*. The position of each fragment has weight 2^j. The value in the degree is 0, 5, 25, ..., $\lceil b/s \rceil$.

Direct conversion of a binary value to a modular one is realized by summing up residues p_i. It is necessary to add all the bit fragments, considering the weights. Thus, a bit binary number with the number of "fragments" *b* can be written as shown below:

$$A = \sum_{j=0}^{\lceil b/s \rceil} \left(\sum_{i=0}^{s-1} x_{j \cdot s+i} 2^i \right) 2^{js},$$

Where *s* is the bit depth; $\lceil b/s \rceil$ – parts (quantity); $x_{j \cdot s+i}$ is a coefficient, usually taken as zero or one.

Based on the above, a formula is obtained that will be used to search for the residue to modulo *p*:

$$|A|_p = \left| \sum_{j=0}^{\lceil b/s \rceil} \left(\sum_{i=0}^{s-1} x_{j \cdot s+i} 2^i \right) 2^{js} \right|_p = \left| \sum_{j=0}^{\lceil b/s \rceil} \left(\left| \sum_{i=0}^{s-1} x_{j \cdot s+i} 2^i \right|_p \cdot \left| 2^{js} \right|_p \right) \right|_p \quad (1)$$

It must be understood that here $\boldsymbol{\omega_i = \left|2^{js}\right|_p}$ are predetermined constants. The residue of the division regarding individual parts can be determined independently of the others. After that, they are all subject to addition to modulo *p*. This approach achieves a smaller bit depth, comparable to the case when one part is considered.

This algorithm is implemented as the following sequence of procedures.

At the stage of preliminary calculations, a two-dimensional table with $\lceil b/s \rceil$ columns is formed. There should be 2^5 lines. Cells with indexes [A_j, j] contain values $\left|A_j \cdot 2^{js}\right|_p$. They must correspond to the different values of A_j and j. They change in the following ranges $j = 0,1, \ldots, \lceil b/s \rceil - 1$; $A_j = 0,1, \ldots, 2^s - 1$. Here, A_j is a number with *s* bits. It is equal to the j-th part of the original. Thus, it is supposed to store $\lceil b/s \rceil \cdot 2^s$values.

When determining a number with *b* bits, cutting into $\lceil b/s \rceil$ is carried out, while the fragments will have dimensions equal to *s*. If the value of *b* cannot be divided by *s* without a remainder, s - (b mod s) of the uppermost bits (necessarily zero ones) is assigned. After that, the values are read from the table. They are used as cell addresses. Summation to modulo *p* is assumed. To do this, it is reasonable to apply recursive doubling. It involves the pairwise addition of all the values obtained to the modulo mentioned. The method provides the distribution of the summation of the latter.

If the number of added values is not even, the unpaired element "leaves" the next layer. The following method can be used to add a pair to the modulo mentioned.

Given the expression (1), A is divided into parts, each of which is 5 bits. The addition is efficient if the fragments are less than modulus p. This is true for the current situation because the tables contain values according to it. The implementation of the approach is described in more detail below.

A search for a constant value U is carried out at the preliminary calculations stage. Its bit depth is $\lceil log_2 p \rceil + 1$. The result is as follows:

$$U = 2^{\lceil log_2 p \rceil} - p.$$

The value of S_1 is interesting at this stage. The bit depth is $\lceil log_2 p \rceil + 1$. The output is:

$$S_1 = A_i + A_j.$$

The case with S_2 should be considered. Its bit depth is similar to the case for S_1. The output is:

$$S_2 = S_1 + U.$$

When the uppermost bit is zero, the sum of the terms in the case of S_1 to modulo p will have a positive value. In a different situation, everything is different. The uppermost bit is excluded. The bit depth, in this case, will be $\lceil log_2 p \rceil$.

Changing the partition parameter s makes it possible to find the remainder of the division with different characteristics. When it comes to magnification, the number of layers to search for the residual part will be less [35,36]. It will reveal an increase in the table formed at the stage of preliminary calculations.

6. Development of a Modified Projection Method with Maximum Likelihood for Error Control When Restoring the Original Representation of Quantities

The process of DDSS Data Reverse Conversion in the direct form (PNS-RRNS) consists of two stages. First, the received fragments are converted from RRNS to PNS. It is supposed to apply methods for detecting and eliminating errors. After that, all parts are connected. The classical scheme in this context implements all the necessary operations. It can be argued that a simplified procedure for detecting and eliminating errors is reasonable for considering the DDSS peculiarities because, in this case, it will be possible to ensure the required performance and reduce the level of resource consumption.

The essence of this simplification is the rejection of the error correction procedure provided for in the classical scheme. When unloading data from DDSS, it is possible to restore the required files by the residues, characterized by the absence of errors, using localization procedures. In post-processing, the representation in RRNS will take place. Thus, the old data with errors will replace the correct ones.

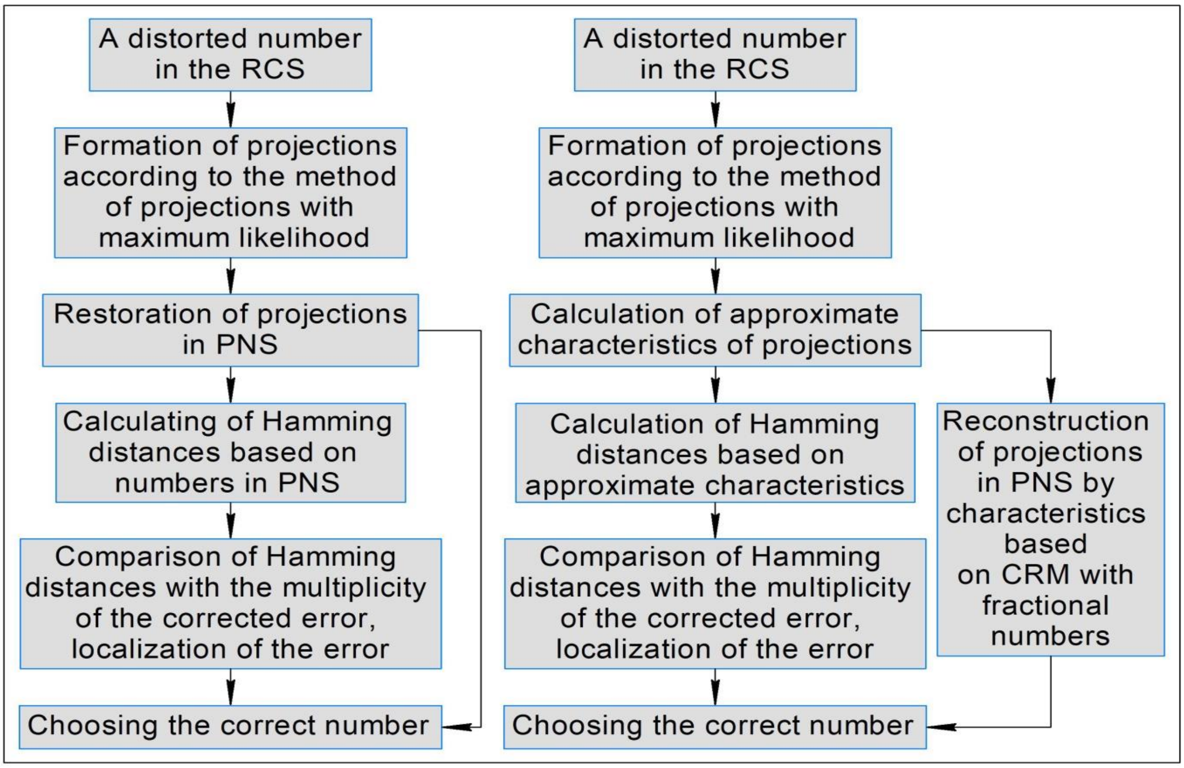

Fig. 5 Schemes of the maximum likelihood projection method (left) and the proposed modification based on the application of the approximate method (right)

In a distributed system, error localization and data recovery can be implemented by applying correction under the condition of a marginal level of likelihood. However, there is computational complexity in this case. Hardware costs are on the rise. It is proposed to use a more efficient modified version of the algorithm shown in Fig. 5. Its essence lies in the approximate calculation of projections and the use of a numerical method to determine the Hamming distance.

7. Results and Discussion

The research aimed to develop novel mathematical models and algorithms based on known alternative technologies, which allow for efficient implementation, of the essential components of DDSS, the use of which will improve system fault tolerance using controlled redundancy.

The essential components of any DDSS are algorithms and methods for representing input data in a distributed form, ensuring their storage and access to them, and reverse recovery of distributed data. Depending on the fault-tolerance mechanisms used, the ways of implementing all these components may be different.

Among the approaches used in practice, the most widely used methods are data replication (creation of backup copies), data separation algorithms, error correcting codes, erasure codes, etc. The features of each method directly influence the cost and the level of complexity of the developed system.

In the process of studies conducted within the framework of the research purpose, it was shown that the DDSS, which is based on the operation of error-correcting code, underpinned with the RRNS, is the most promising structure for a distributed fault-tolerant storage system. The results of the analysis of other traditional ways of organizing DDSS, some of which are illustrated in Fig. 6 and 7, confirmed the validity of this solution.

Thus, the advantage of replication is the guarantee of full accessibility and integrity of stored information. But this easiest way to create backup data is associated with increased redundancy of DDSS and rather high storage costs. Data partitioning algorithms are associated with the expenditure of significant computing resources and a wide spectrum of error-correcting codes. Erasure codes are also characterized by increased complexity. Although, due to the existing methods for simplifying their implementation, DDSS, which is based on this method for maintaining increased fault tolerance, is one of the most effective from technical and economic viewpoints.

Another significant disadvantage of all methods listed above is their low performance when it is necessary to process data in a distributed form and ensure the safety and security of stored information from leaks. The most promising solution to this problem is using multifunctional error-correcting codes compatible, particularly with arithmetic coding. These possibilities are inherent in RNS and RRNS-based codes, which have found application as a non-positional notation and intend for organizing computational operations with data that exists in a distributed form.

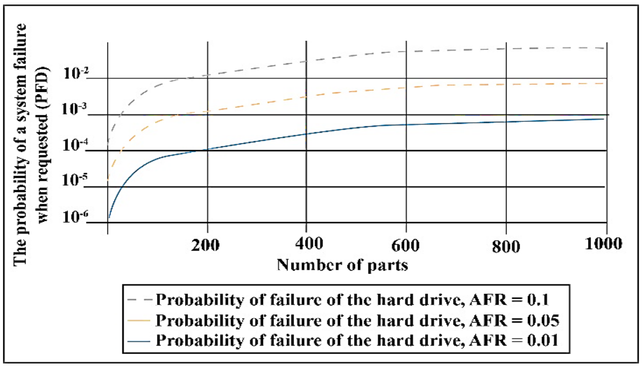

Fig. 6 Probability of failure of distributed storage system based on redundant RNS of type (2, 6) at error probability *er* = 0.001, after T0 = 8766 hours

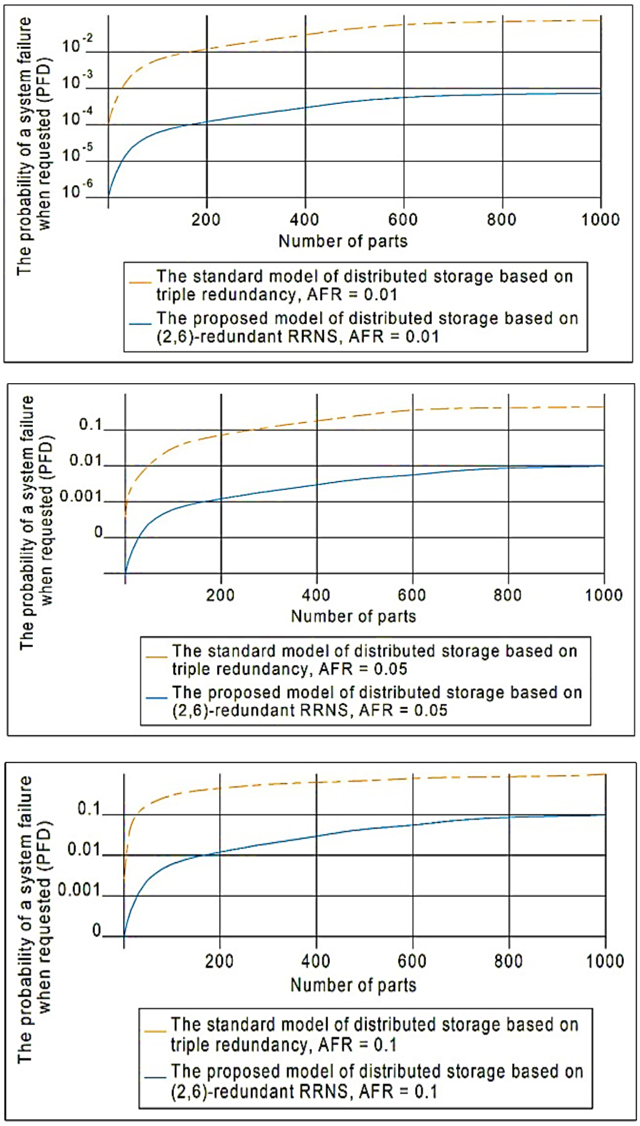

Fig. 7 Probability of failure of distributed storage systems using triple redundancy and type (2, 6) partitioning scheme at *er* = 0.001, T0 = 8766 hours. (a) AFR = 0.01. (b) AFR = 0.05. (c) AFR = 0.1.

Among other things, RRNS can detect arithmetic processing errors caused by problems such as device noise, manufacturing defects, or changes in device operation. However, its obvious disadvantage is the need for a large computational time in the error-detecting and correcting processes. A similar problem is explained by the presence of iterative calculations required in the sequential determination of the erroneous digits of the residue.

Based on the previous, it can be concluded that it is advisable to improve methods for detecting and correcting errors and tools for improving fault tolerance and reliability. In this case, fault-tolerant algorithms for detecting errors and recovering correct data using their direct and inverse transformation in distributed systems with modular arithmetic should be considered the most promising vector of such development.

The use of effective methods from the viewpoint of hardware costs is a way to solve the presented problem: to implement changes in the images of distribution subsystems and projections with a maximum level of likelihood to monitor errors in the restoration of "a priori" forms of certain parameters.

8. Conclusion

Technologies that allow for distributed data storage organization are increasingly mastered to increase the fault tolerance of storage systems for ever-increasing volumes of information in various areas of human activity.

The work done allows us to state that many advanced approaches ensuring the stability of distributed systems to failures assume the formation of redundancy. There is complexity. However, data backup seems to be optimal. It should be understood that erasure codes are the most complex option in the algorithmic aspect. Correcting codes is an alternative in both cases. The DDSS, based on the operation of error-correcting code underpinned with the RRNS, is the most promising structure for a distributed fault-tolerant storage system.

It has been analytically proven that the projection method with the maximum level of likelihood is optimal in this context. However, this method has a high computational complexity, which can significantly affect the hardware costs and DDSS performance. It is proposed to use a more efficient modified version of this method, which implements an approximate calculation of projections, and applies a numerical method to determine the Hamming distance.

Considering that a specific error in a particular DDSS element can provoke an incorrect recovery result, it can be said that it is reasonable to improve methods for finding, localizing and eliminating errors. It is essential because, in the current environment, any application's use of data in DDSS results in delays against the background of existing approaches.

The experience of advanced countries shows that RRNS improves fault tolerance and reduces storage redundancy. However, simultaneously, one should note the insufficiently intensive development of collaborative technologies with high efficiency for error correction, the use of RRNS, and artificial neural networks.

Author Contributions

AAT is responsible for conceptualization, reviewing and editing the manuscript draft and supervising and administering the project. AKL conducted a formal analysis and investigation and prepared the manuscript draft. DVP is in charge of the methodology and software. RKT carried out the validation, data curation, and visualization. All authors read and approved the manuscript.

Funding Statement

Selected findings of this work were obtained under the Grant Agreement in the form of subsidies from the federal budget of the Russian Federation for state support for establishing and developing world-class scientific centers performing R&D on scientific and technological development priorities dated April 20, 2022, No. 075-15-2022-307.

References

[1] Conor JohnCremin et al., “Big Data: Historic Advances and Emerging Trends in Biomedical Research,” *Current Research in Biotechnology,* vol. 4, pp. 138–151, 2022. *Crossref,* https://doi.org/10.1016/j.crbiot.2022.02.004

[2] Manar Sais, Najat Rafalia, and Jaafar Abouchabaka, “Intelligent Approaches to Optimizing Big Data Storage and Management: REHDFS System and DNA Storage,” *Procedia Computer Science*, vol. 201, pp. 746–751, 2022. *Crossref,* https://doi.org/10.1016/j.procs.2022.03.101

[3] Sarah Wordsworth et al., “Using “Big Data” in the Cost-Effectiveness Analysis of Next-Generation Sequencing Technologies: Challenges and Potential Solutions,” *Value in Health,* vol. 21, no. 9, pp. 1048–1053, 2018. *Crossref,* https://doi.org/10.1016/j.jval.2018.06.016

[4] Abas H. Lampezhev, Islam A. Alexandrov, and Victor A. Gorelov, “Automated Analysis of Big Data From Social Networks as a Way to Compile a Psychological Portrait of a Personality,” *Proceedings of the International Conference on Quality Management, Transport and Information Security, Information Technologies,* pp. 511-515, 2021. *Crossref,* https://doi.org/10.1109/ITQMIS53292.2021.9642901

[5] NataliyaShakhovska et al., “Big Data Processing Technologies in Distributed Information Systems,” *Procedia Computer Science*, vol.160, pp. 561–566, 2019. *Crossref,* https://doi.org/10.1016/j.procs.2019.11.047

[6] Mohammad S. Aslanpour, Sukhpal Singh Gill, and Adel N. Toosi, "Performance Evaluation Metrics for Cloud, Fog and Edge Computing: A Review, Taxonomy, Benchmarks and Standards for Future Research," *Internet of Things,* vol.12, 100273, 2020. *Crossref,* https://doi.org/10.1016/j.iot.2020.100273

[7] Dalia Kamal A. A. Rizk et al., "Applying Ai for Timely Input to a Smart Healthcare System," *Journal of Southwest Jiaotong University,* vol. 57, no. 4, pp. 312–325, 2022. *Crossref,* https://doi.org/10.35741/issn.0258-2724.57.4.28

[8] HishamAl-Ward, Chee Keong Tan, and Wern Han Lim, "Caching Transient Data in Information-Centric Internet-of-Things (IC-IoT) Networks: A Survey," *Journal of Network and Computer Applications,* vol. 206, p. 103491, 2022.

[9] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller, "Methods for Interpreting and Understanding Deep Neural Networks," *Digital Signal Processing,* vol. 73, pp. 1–15, 2018. *Crossref,* https://doi.org/10.1016/j.dsp.2017.10.011

[10] YongliangXu et al., "Secure Fuzzy Identity-Based Public Verification for Cloud Storage," *Journal of Systems Architecture,* vol.128, p. 102558, 2022. *Crossref,* https://doi.org/10.1016/j.sysarc.2022.102558

[11] MuntadherSaadoon et al., "Fault Tolerance in Big Data Storage and Processing Systems: A Review on Challenges and Solutions," *Ain Shams Engineering Journal*, vol. 13, no. 2, p. 101538, 2022.

[12] Aslan Tatarkanov et al., "Suboptimal Biomedical Diagnostics in the Presence of Random Perturbations in the Data," *International Journal of Engineering Trends and Technology,* vol. 70, no. 11, pp. 129–137, 2022. *Crossref,* https://doi.org/10.14445/22315381/IJETT-V70I11P213

[13] Anan Zhou, Benshun Yi, and Laigan Luo, "Tree-Structured Data Placement Scheme with Cluster-Aided Top-Down Transmission in Erasure-Coded Distributed Storage Systems," *Computer Networks,* vol. 204, p. 108714, 2022. *Crossref,* https://doi.org/10.1016/j.comnet.2021.108714

[14] Islam Alexandrov et al., "Development of Algorithm for Calculating Data Packet Transmission Delay in Software-Defined Networks," *Emerging Science Journal,* vol. 6, no. 5, pp. 1062–1074, 2022. *Crossref,* https://doi.org/10.28991/ESJ-2022-06-05-010

[15] Wenqi Cao, and Cong Zhang, "An Effective Parallel Integrated Neural Network System for Industrial Data Prediction," *Applied Soft Computing,* vol. 107, p. 107397, 2021. *Crossref,* https://doi.org/10.1016/j.asoc.2021.107397

[16] M. Szymczyk, and P. Szymczyk, "Automatic Processing of Z-Transform Artificial Neural Networks Using Parallel Programming," *Neurocomputing,* vol. 379, pp. 74–88, 2020. *Crossref,* https://doi.org/10.1016/j.neucom.2019.10.078

[17] BinLiao et al., "Energy-Efficient Algorithms for Distributed Storage System Based on Block Storage Structure Reconfiguration," *Journal of Network and Computer Applications,* vol. 48, no. 1, pp. 71–86, 2015. *Crossref,* https://doi.org/10.1016/j.jnca.2014.10.008

[18] Yijie Wang, and Sijun Li, "Research and Performance Evaluation of Data Replication Technology in Distributed Storage Systems," *Computers & Mathematics with Applications,* vol. 51, no. 11, pp. 1625–1632, 2006. *Crossref,* https://doi.org/10.1016/j.camwa.2006.05.002

[19] E. V. Sokolov, and E. V. Kostyrin, "Breakthrough Technologies Financing of Development of Science Researches and Competitive Medical Equipment," *AIP Conference Proceedings,* vol. 2250, p. 020026, 2020. *Crossref,* https://doi.org/10.1063/5.0013333

[20] Quanlu Zhang et al., "Ustore: A Low Cost Cold and Archival Data Storage System for Data Centers," *Proceedings of the 35th International Conference on Distributed Computing Systems,* pp. 431–441, 2015. *Crossref,* https://doi.org/10.1109/ICDCS.2015.51

[21] Jianjiang Li et al, "A Data-Check Based Distributed Storage Model for Storing Hot Temporary Data," *Future Generation Computer Systems,* vol. 73, pp. 13–21, 2017. *Crossref,* https://doi.org/10.1016/j.future.2017.03.019

[22] Alexander Thomasian, "*Storage Systems. Organization, Performance, Coding, Reliability, and Their Data Processing,*" Burlington, Vermont, United States: Morgan Kaufmann, pp. 89-196, 2022.

[23] Lluis Pamies-Juarez, Anwitaman Datta and Frédérique Oggier, "In-Network Redundancy Generation for Opportunistic Speedup of Data Backup," *Future Generation Computer Systems,* vol. 29, no. 6, pp. 1353–1362, 2013. *Crossref,* https://doi.org/10.1016/j.future.2013.02.009

[24] Dongqing Wang, Feng Ding, and Yanyun Chu, "Data Filtering Based Recursive Least Squares Algorithm for Hammerstein Systems Using the Key-Term Separation Principle," *Information Sciences,* vol. 222, pp. 203–212, 2013. *Crossref,* https://doi.org/10.1016/j.ins.2012.07.064

[25] KaiSong et al., "Research and Application of Error Correction Theory for Ternary Optical Computer Based on Hamming Code," *Optik,* vol. 267, 169647, 2022. *Crossref,* https://doi.org/10.1016/j.ijleo.2022.169647

[26] J. Wang, H. Wu, and R. Wang, "A New Reliability Model in Replication-Based Big Data Storage Systems," *Journal of Parallel and Distributed Computing,* vol.108, pp. 14–27, 2017. *Crossref,* https://doi.org/10.1016/j.jpdc.2017.02.001

[27] P. Shah, and R. Oza, *Information and Communication Technology for Intelligent Systems (ICTIS 2017),* vol. 2, 2017. *Smart Innovation, Systems and Technologies,* S. Satapathy, and A. Joshi, Eds. Cham, Switzerland: Springer, vol.84, pp. 236-244, 2018.

[28] Alessandro Neri, "Twisted Linearized Reed-Solomon Codes: A Skew Polynomial Framework," *Journal of Algebra*, vol. 609, pp. 792–839, 2022. *Crossref,* https://doi.org/10.1016/j.jalgebra.2022.06.027

[29] Han Bao, Yijie Wang, and Fangliang Xu, "Reducing Network Cost of Data Repair in Erasure-Coded Cross-Datacenter Storage," *Future Generation Computer Systems,* vol. 102, pp. 494–506, 2020. *Crossref,* https://doi.org/10.1016/j.future.2019.08.027

[30] Victor A. Gorelov et al., "Complex Methodological Approach to Introduction of Modern Telemedicine Technologies into the Healthcare System on Federal, Regional and Municipal Levels," *2020 International Conference Quality Management, Transport and Information Security, Information Technologies (IT&QM&IS)* pp. 468-473, 2020. *Crossref,* https://doi.org/10.1109/ITQMIS51053.2020.9322864

[31] AlanPinheiro et al., "Optimized Buffer Protection for Network-on-Chip Based on Error Correction Code," *Microelectronics Journal*, vol. 100, p. 104799, 2020. *Crossref,* https://doi.org/10.1016/j.mejo.2020.104799

[32] N. I. Chervyakov et al., "Residue Number System-Based Solution for Reducing the Hardware Cost of a Convolutional Neural Network," *Neurocomputing,* vol. 407, pp. 439–453, 2020.

[33] Avik Sengupta, and Balasubramaniam Natarajan, "Redundant Residue Number System Based Space-Time Block Codes," *Physical Communication*, vol.12, pp. 1–15, 2014. *Crossref,* https://doi.org/10.1016/j.phycom.2014.01.002

[34] E. V. Kostyrin, "The Economic and Mathematical Model of Medical Organization Management," *2020 13th International Conference Management of Large-Scale System Development-(MLSD),* p. 9247652, 2020. *Crossref,* https://doi.org/10.1109/MLSD49919.2020.9247652

[35] Rimma Meyramovna Ualiyeva et al., "Peculiarities of the Structure of Male Reproductive System in Trematode Parastrigea Robusta (Trematoda: Strigeidae)," *Online Journal of Biological Sciences,* vol. 17, no. 2, pp. 88–94, 2017. *Crossref,* https://doi.org/10.3844/ojbsci.2017.88.94

[36] Rimma Meyramovna Ualiyeva, Sayan Berikovich Zhangazin, and Indira Bulatovna Altayeva, "Structural Organization of Vitelline Cells of Trematode with Undifferentiated Body of Azygia Lucii (Muller, 1776)," *Online Journal of Biological Sciences,* vol. 22, no. 1, pp. 10–17, 2022. *Crossref,* https://doi.org/10.3844/ojbsci.2022.10.17