THE RELIABILITY MODEL OF A DISTRIBUTED DATA STORAGE IN CASE OF EXPLICIT AND LATENT DISK FAULTS

Lyudmila Ivanichkina and Andrew Neporada
OOO Proekt IKS, Altufievskoesh, Moscow, Russia
E-Mail: livanichkina@parallels.com

ABSTRACT

This work examines the approach to the estimation of the data storage reliability that accounts for both explicit disk faults and latent bit errors as well as procedures to detect them. A new analytical math model of the failure and recovery events in the distributed data storage is proposed to calculate reliability. The model describes dynamics of the data loss and recovery based on Markov chains corresponding to the different schemes of redundant encoding. Advantages of the developed model as compared to classical models for traditional RAIDs are covered. Influence of latent HDD errors is considered, while other bit faults occurring in the other hardware components of the machine are omitted. Reliability is estimated according to new analytical formulas for calculation of the mean time to failure, at which data loss exceeds the recoverability threshold defined by the redundant encoding parameters. New analytical dependencies between the storage average lifetime until the data loss and the mean time for complete verification of the storage data are given.

Keywords: mean time to failure, Markov chains, redundant encoding, Huygens' gambler's ruin problem, distributed data storage, scrubbing procedure, checksums, MTTDL of distributed data storage, disk faults, irrecoverable bit errors, latent sector errors.

1. INTRODUCTION

Petabyte size data storages are composed of a large number of hard disk drives. Data integrity in a distributed system largely depends on reliability and performance characteristics of HDDs used. Determining the storage reliability in general is an important applied problem that becomes more complex as the number of disks in the storage grows.

A hard disk, which is a complex technical device, is prone to failures that are caused by a set of various random factors and are of stochastic nature. With a certain level of accuracy, probability of HDD failure can be approximated by statistical law, then the fault occurrence process can be generalized to the whole population of disks in the storage.

A model of storage reliability describing a series of HDD faults and replacements prior to a certain condition becomes true corresponds to a Huygens' gambler's ruin problem. The Huygens class of problems considers a game with the limited initial sum at the fixed stake and the known math expectation of winning in each round. This game is played either till the initial capital is augmented or till it is completely lost. Two equivalent approaches to this problem are widely used, which are: Bernoulli process and random walk. Thus, the random walk method considers the stochastic movement of the gambler among discrete states depending on the outcome of another round.

In the storage reliability model, assuming all disks are equally significant and independent, the number of discrete states corresponds to the number of operational HDDs. Transition between states is defined by explicit failure detected right after its occurrence. In particular, the fault of the HDD firmware is one of such states.

Since backup methods are used in actual applications, data integrity is preserved when a certain portion of HDDs fail. The maximum threshold of failed HDDs is defined by the scheme and parameters of redundant encoding, and if exceeded, causes irrecoverable data loss.

Another technical aspect of HDDs is the occurrence of latent bit errors that are not detected right at the moment they occur. Bit errors corrupt the stored data but do not affect the physical operation of equipment in any way. Since the recovery procedure is started only when errors are detected, latent bit errors have negative impact on the storage reliability. This class of errors belongs to irrecoverable read errors and should be considered when designing data storages.

Calculation of checksums for the fragments of the each data block with the subsequent verification of these checksums is the main way to deal with such errors. Several approaches to data verification are possible.

According to the first method, checksums are verified only when the client requests the data. The recovery process is initiated if a checksum mismatch is detected. In this case, the mean intensity of data access is an important parameter affecting the storage reliability. In practice, the intensity of access to different types of data may vary significantly, so risk of losing data for a large archive file that is rarely accessed may be rather high. Bearing this in mind, we can state that the first option does not provide the sufficient level of storage reliability for all files.

The second approach to checksum verification presumes that the storage has a centralized service that manages the continuous process of the checksum verification for the data fragments. This technique is also known as scrubbing. In this method, the centralized service does not necessarily have to recalculate checksums. It is sufficient for this service to implement storage of information required to check data, and to send data verification commands to the storage services. Thus, the load related to the data verification process is distributed across the entire storage. The intensity of such a process is usually estimated by the mean scrubbing interval, i.e. the expected time to check the entire data contained in the storage that varies from several days to several weeks.

This work proposes a new math model of storage reliability with original analytical description of transitions between states. The model gives consideration to latent disk errors and continuous scrubbing process. The presented model builds on the ideas of describing traditional hardware of local data storage in RAIDs of individual HDDs and considers advanced distributed data storage systems with redundant encoding where individual fragments of data blocks are not bound to disks and are moved due to the replication or load balancing processes. The developed math model redefines the semantics of classical models of the RAID systems – their states correspond to the block fragments rather than to the disks. Similarly, transitions between the states describe loss or recovery of a data fragment, rather than failure of an individual HDD.

2. LITERATURE REVIEW

Classical data storage reliability models based on Markov chains in continuous time provide approximate idea about the storage MTTDL but do not account for bit errors on disks. These models are covered in several works (for example [1], [2], [3], [4]) dedicated to reliability of RAIDs in conditions of explicit disk failures. In recent years, an increasing number of scientific works is dedicated to the development of the extended Markov reliability models that include math descriptions of latent disk errors and processes used to detect them. Thus, for example, drawbacks of classical models with unidimensional Markov chains without memory are described in details in the well-known work [5]. Counterarguments showing the applicability of Markov chains for reliability estimation are provided in [6], where the memory effect is reproduced within more accurate detailed description of the system by increasing the number of states and transitions between them. The original model for a RAID of SSDs with growth of error intensity according to amortization is proposed in [7]. For example, [8, 9, 10, 11] and [12] propose the consistent generalizations of the RAID systems reliability model considering latent sector read errors and mechanisms of their detection (scrubbing).However, the underlying RAID-5 and RAID-6 data redundancy schemes result in severe limitations to governing mathematical models. The RAID-groups usually allow only the complete scanning of the whole disks. This mode of operation is unsuitable for data storages, consisting of high-capacity disks due to prolonged scrubbing times and wasteful free-space checking of half-empty disks [13]; [14]. Moreover, in [15] Iliadis*et al.*state that the excessive scrubbing rate decreases reliability. To overcome the mentioned problems, in [16] Liu *et al.* propose a frequency-cost function to keep an optimum trade-off between the reliability and the data cost by adjusting dynamically the scrubbing frequency.In [17]Venkatesan*et al.* summarize the effect of latent errors stating that the ones of high probability reduce the reliability of RAID-6 to that of RAID-5 without sector errors for all symmetric data placement schemes and all MDS erasure codes.

In published works, the two-dimensional Markov model, due to its complexity, is examined by numeric simulation methods that include statistical modeling. Despite complexity of describing two-dimensional models, Markov models remain attractive to use, since they providean estimation of the data storage reliability for an arbitrary scheme of redundant encoding with consideration given to various types of disk failures, different states of storage components, and various policies for replacement of failed components and data recovery, as well as varying intensities of recovery for a different number of lost fragments. Unlike simulation modeling based on Monte-Carlo statistical tests, this method allows obtaining of exact analytical expression for storage reliability within the given model. This analytical expression may be used for identification of certain functional dependencies between the model parameters. Besides, Markov models preserve significant advantage in terms of performance and calculation speed (up to 150 times, see [6]) as compared to the full scale simulation.

3. METHODOLOGY

a) Basic Math model

Assume there is a data block that consists of n encoded fragments, of which k fragments correspond to the source data, while the remaining$(n-k)$ fragments are checksums. Also, assume that the encoding scheme allows data to be recovered when any k fragments are available. A data block is defined by the following set of states S_i: S_0 - all n fragments are available, and no one fragment is defective, S_1 - one fragment is defective, S_2 - 2 fragments are defective, …, S_{n-k} - $(n-k)$ fragments are defective, S_{n-k+1} - more than $(n-k)$ fragments are defective, which means it is impossible to recover the block, i.e., the block data are lost. For future use, the $(n-k+1)$ state is convenient to call DL (“data loss”) to stand it out among the remaining states. The DL state is absorbing one, since there in no back transitions into other states of the system for it. According to the model, in certain time the system will go into the absorbing state regardless of its initial state with probability of 1. Important practical characteristic of such systems is the mean time of operation until transition into the absorbing state.

The system's transition between states describes loss of data fragments in a block due to HDD failures. As the first simplified assumption, we can say that disk failures comply with Poisson distribution, according to which probability of a disk failure during any given period of time does not depend on the disk age. Disk failure means complete loss of the data stored on the disk, and so the lost fragments can be restored only using the fragments residing on operational disks.

Assume, the intensity of disk failures is λ, which means that the expected time to failure of a disk is $1/\lambda$. For the majority of HDD models failure the intensity is about 10^{-9} second. Probability of disk failure within the $[0, t)$ time interval is defined by the disk failure intensity and is $\Pr(t_F < t) = 1 - e^{-\lambda t}$.

The second generally accepted assumption is the supposition about the mutual independence of individual disk failures. In this case, for n working disks the total failure intensity is $n\lambda$, while the expected mean time to failure of at least one disk of n is $1/(n\lambda)$.

A disk failure defines transition of the model from the S_l state with l non-functional disks into the S_{l+1} state with $(l + 1)$ non-functional disks. Since in models with continuous time simultaneous failure of any two disks is assumed to have zero probability, it may be considered that there are no direct transitions due to the failures between the states that differ by more than one operational disk.

The next important factor of the model is the mean time to data fragment recovery$1/\mu$. Unlike disk failure intensity, this parameter is defined not only by physical properties of the disk, but also by the architecture of a certain distributed storage. It also may depend on many other factors. The mean time to recover data fragment is made up of the time from the moment of disk failure to its detection and the time from the failure detection to the moment when the fragment to be recovered is written on the one of the disks of the system. Generally, the architecture of distributed data storage provides a monitoring service that tracks the state of all blocks in the data storage and launches the data recovery procedure when failures are detected. Because of it, the time to failure detection, even for large volumes of data stored in the system, can be considered small enough - about several minutes. Below, the classical model is generalized to the scenario of occurring latent disk failures and the procedures to detect them.

b) Extended model

Frequency of irrecoverable read errors

Generally, frequency of irrecoverable read errors is provided by the manufacturer in the disk specification and is about one error per 10^{14} bits or about one error per 11 TB of data read from the disk. To account for this parameter in the proposed model, we need to estimate how often such errors occur. To do this, we need to estimate the average amount of data read from a single disk in the data center during some characteristic time – a year, for example.

The mean disk access intensity can be estimated based on the average disk load, provided in the specification. Assume that the expected disk load is about 20%, i.e., during 20% of the time the disk is accessed and it is idle during 80% of time. Then, the steady-state speed of sequential disk read is 140 Mbps, according to this, $(140 \cdot 0.2 \cdot 3600 \cdot 24 \cdot 365)/1024 = 862312$ GB or about 842 TB is read from the disk annually. Based on this, frequency of irrecoverable bit errors for a single disk by order of magnitude can be estimated as about 77 errors per year. Assume the disk size is 2 TB and it is half full, while the typical size of a data fragment in the storage is 50 MB. Then, the mean frequency of read errors for the selected data block fragment is about 0.00367 errors per year or about 1 error during 272 years.

The parameter c (frequency of irrecoverable read error for a single data block fragment) that in this case is 272years^{-1} has been added into the model. The value of this parameter can be compared to the disk failure intensity that for disks with MTTF=200000 hours can be estimated as $23\,\text{years}^{-1}$. This circumstance should be accounted for when evaluating MTTDL of the data storage.

It is worthy of note that this parameter was calculated for the scenario of relatively high degree of disk usage. In general, frequency of irrecoverable read errors depends significantly on disk usage intensity. The average expected disk load defined in the specification is a value for approximate estimates. However, for certain data storage it is better to take the average estimates of disk usage for the given storage. For example, archive data storage may have lower disk usage rate. In general, frequency of irrecoverable read errors depends on the data access intensity and patterns of the actual data center. However, the calculated value of this parameter will provide the lower-bound estimate of the storage MTTDL based on presence of bit errors and the way they affect data storage reliability.

The mean scrubbing interval

The mean scrubbing interval, i.e.the mean time to perform complete data integrity check in a data center, is defined by the mean data verification intensity i.e., the average amount of data checked in a unit of time. Physically realistic values for the mean data integrity checking intensity depend on the storage load level and guaranteed performance the storage is required to provide to its clients, as well as economic costs to perform scrubbing.

Depending on the storage load level user tasks and system processes may compete for the same resources. The examples of such resources in a distributed storage system are CPU time, disk IO operations and network throughput. In this case, as a rule, resources used by service processes are limited to provide customers with the guaranteed performance level (guaranteed response time, guaranteed number of the IO operations per second).

In cases when the actual storage load level is far from the maximum values and system processes don't interfere with the user queries the data integrity checking intensity is completely defined by the balance between the data storage reliability requirements and the maximum allowed costs for the scrubbing process. Firstly, these costs are defined by the possible necessity to start spinning the disk containing the fragment to be verified in case when the disk is in the standby mode and its power consumption is low. Secondly, costs depend on the need to use CPU time for the computationally intensive algorithm of checksum calculation as well as, possibly to wake one of CPU cores up from the low power consumption mode.

As an approximate value for the mean scrubbing interval, a period specific for the data center, from one week to one month, can be selected. Also, the mean time of complete verification should be less than the mean time between requests to the same data by the storage user. The value of data integrity checking intensity is α.

In a classical model, the state of a distributed storage system is fully defined by the number of faulty HDDs. In the new model, the system state depends not only on the number of the explicit, but also on the number of the latent fragment faults. So, the extended model, unlike the classical model, is built on two-dimensional rather than unidimensional space of the Markov chain states.

Assume that the state of the (l, m) system is defined by explicit faults of l disks and by m latent fragment corruptions. Whereby, latent fragment corruptions are accounted only for operational disks, since data on faulty disks are considered unavailable.

It also should be noted that the value of the sum $(l + m)$ for the state, in which block data still can be restored is limited on top by the number of fragments that can be lost in this error-correcting code without losing data. In other words, the $0 \leq l + m \leq n - k$ inequality is true for this state. For states with the $l + m > n - k$ condition, block data are irrecoverable, therefore, regardless of the ratio of latent and explicit faults, all these states may be joined into a single DL state ("data loss").

So, the total number of states in the system, apart from the DL state, is

$$\sum_{i=1}^{n-k+1} i = (n-k+1)\frac{n-k+2}{2}.$$

The first type of state transition describes disk failures. Two transition options are possible for the (l,m) state. Firstly, if a disk corresponding to one of the corrupted fragments fails then the system with intensity of mλ goes into the (l+1, m-1) state. Secondly, if a disk corresponding to one of the intact fragments fails then the system with intensity of (n-l)λgoes into the (l+1,m) state.

The second transition type is transition into states with latent corruptions because of the bit errors. Bit errors can cause transition from the (l, m) state into the $(l, m + 1)$ state with intensity of $(n - l - m)c$. This transition does not account for latent bit errors on disks that already failed and also does not consider the possibility of new bit errors for already corrupted fragments since these events do not change the state of the system.

Data recovery related transitions between the states take place due to the recovery process initiated after an explicit disk fault or latent bit corruption detection. Assume that when recovering after explicit faults, checksums for fragments on operational disks are verified, and conversely, when latent errors are detected not only the corrupted fragments are rewritten, but also fragments lost as a result of explicit disk failures are recovered. Also, assume that the data recovery process always moves the system into the $(0, 0)$ state without any explicit or latent faults. The move into the $(0, 0)$ state is based on the fact that the MTTDL value for the sequential and parallel data recovery processes is the same in the case when intensity of recovery processes is much higher than intensity of disk failures. Depending on the state of the system, intensity of transition into the $(0, 0)$ state changes as follows: for states of the $(l, 0), l > 0$ type, intensity is μ, for states of the $(0, m), m > 0$ type, intensity is α, and for states of the $(l, m), l > 0, m > 0$ type, intensity is $(\alpha + \mu)$.

In the new model, transitions into the DL state differ from transitions into other states, since the DL state combines multiple states. In the DL state, the system may shift from states of the $(i, n - k - i)$ type, where $0 \leq i \leq n - k$. Intensity of the transition into the DL state for any of these states is $k(\lambda + c)$.

MTTDL calculation with account for irrecoverable bit errors and scrubbing process

Prior to the description of the model with arbitrary n and k parameters, for illustration purposes, it is helpful to consider practically important special cases of MTTDL calculation with values $n - k = 1$and $n - k = 2$. Applied efficiency of these parameter sets for Locally Repairable Codes (LRC) is confirmed in [18]and [12]. The schemes of the system states and transitions between them for these cases are shown on Figure-1 and Figure-2. In general, the calculation algorithm is similar to calculations for the classical model.

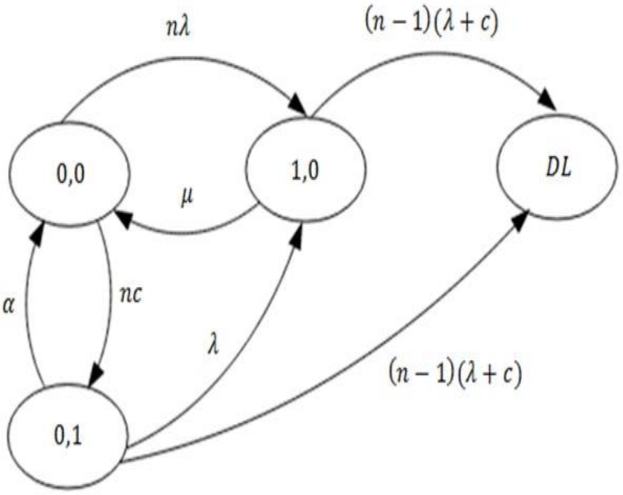

Figure-1. Intensity of transitions in the Markov chain for the case of n-k=1.

Assume that $Q_{l,m}$ is the probability of the system moving from the initial (l, m) state into the DL state prior to getting into the $(0, 0)$ state without any latent or explicit failures. The $Q_{l,m}$ definition suggests that $Q_{0,0} = 0$, and $Q_{DL} = 1$.

The value $Q_{1,0}$ is expressed through $Q_{0,0}$ and Q_{DL} probabilities for the states, in which the system can move from the $(1,0)$ state:

$$Q_{1,0} = \frac{1}{\mu + (n-1)(\lambda + c)}\left(\mu Q_{0,0} + (n-1)(\lambda + c)Q_{DL}\right).$$

From this item on when performing calculations we assume that values of λ and c parameters are negligible as compared to values of μ and α parameters. Given that $Q_{0,0} = 0$, $Q_{DL} = 1$ and omitting negligible elements, we can obtain:

$$Q_{1,0} \approx \frac{(n-1)(\lambda + c)}{\mu}.$$

Value of $Q_{0,1}$is expressed as follows:

$$Q_{0,1} = \frac{1}{\alpha + \lambda + (n-1)(\lambda + c)}\left(\alpha Q_{0,0} + \lambda Q_{1,0} + (n-1)(\lambda + c)Q_{DL}\right).$$

We can note that the element $\lambda Q_{1,0} \sim \lambda(\lambda + c)$, and the element $(n-1)(\lambda + c)Q_{DL} \sim (\lambda + c)$, consequently, $\lambda Q_{1,0}$ are negligible in relation to $(n-1)(\lambda + c)Q_{DL} \sim (\lambda + c)$:

$$Q_{0,1} \approx \frac{(n-1)(\lambda + c)}{\alpha}.$$

In this model, two alternatives of the cycle start are possible; the cycle can start either from a disk failure, i.e., from the $(1,0)$ state, or from irrecoverable read error, i.e., from the $(0,1)$ state.

The expected number of cycles to data loss in case of starting from the $(1,0)$ state is

$$n_{e,(1,0)} = \frac{1}{Q_{1,0}} \approx \frac{\mu}{(n-1)(\lambda + c)}.$$

The expected number of cycles to data loss in case of starting from the $(0,1)$ state is

$$n_{e,(0,1)} = \frac{1}{Q_{0,1}} \approx \frac{\alpha}{(n-1)(\lambda + c)}.$$

The mean cycle time is calculated as follows. Assume that $T_{l,m}$ is expected time that will pass prior to the system gets from the (l,m) state into the $(0,0)$ orDL state. By definition, $T_{0,0} = 0$, $T_{DL} = 0$.

Values of $T_{l,m}$for the system states are calculated as

$$T_{1,0} = \frac{1}{\mu + (n-1)(\lambda + c)}\left(\mu T_{0,0} + (n-1)(\lambda + c)T_{DL}\right) + \frac{1}{\mu + (n-1)(\lambda + c)};$$

$$T_{1,0} = \frac{1}{\mu + (n-1)(\lambda + c)};$$

$$T_{1,0} \approx \frac{1}{\mu};$$

$$T_{0,1} = \frac{1}{\alpha + \lambda + (n-1)(\lambda + c)}\left(\alpha T_{0,0} + \lambda T_{1,0} + (n-1)(\lambda + c)T_{DL}\right) + \frac{1}{\alpha + \lambda + (n-1)(\lambda + c)};$$

$$T_{0,1} = \frac{1}{\alpha + \lambda + (n-1)(\lambda + c)}\frac{\lambda}{\mu} + \frac{1}{\alpha + \lambda + (n-1)(\lambda + c)};$$

$$T_{0,1} \approx \frac{1}{\alpha + \lambda + (n-1)(\lambda + c)} \approx \frac{1}{\alpha}.$$

Let$t_{0,0\to 1,0}$ and $t_{0,0\to 0,1}$be mean times to shifting from the $(0,0)$ state into the $(1,0)$ and$(0,1)$ states respectively.

For mean cycle times $t_{e,(1,0)}$ and $t_{e,(0,1)}$ in case of starting from the $(1,0)$ and$(0,1)$ states the following formulas are true:

$$t_{e,(1,0)} = t_{0,0\to 1,0} + T_{1,0};$$
$$t_{e,(0,1)} = t_{0,0\to 0,1} + T_{0,1};$$
$$t_{e,(1,0)} \approx \frac{1}{n\lambda} + \frac{1}{\mu};$$
$$t_{e,(0,1)} \approx \frac{1}{nc} + \frac{1}{\alpha}.$$

Given that the $1/\mu$ and $1/\alpha$ values are negligible as compared to$1/(\lambda + c)$, we can obtain

$$t_{e,(1,0)} \approx \frac{1}{n\lambda}$$

$$t_{e,(0,1)} \approx \frac{1}{nc}.$$

The above formulas suggest that storage MTTDL estimation for the new model is equal to the least value of the expected times to data loss in case of starting from the $(1,0)$ or$(0,1)$ state:

$$MTTDL_{n-k=1} = \min\left(n_{e,(1,0)}t_{e,(1,0)}, n_{e,(0,1)}t_{e,(0,1)}\right) \approx \min\left(\frac{\mu}{n\lambda(n-1)(\lambda+c)}, \frac{\alpha}{nc(n-1)(\lambda+c)}\right).$$

Next we examine the $n-k=2$ case.

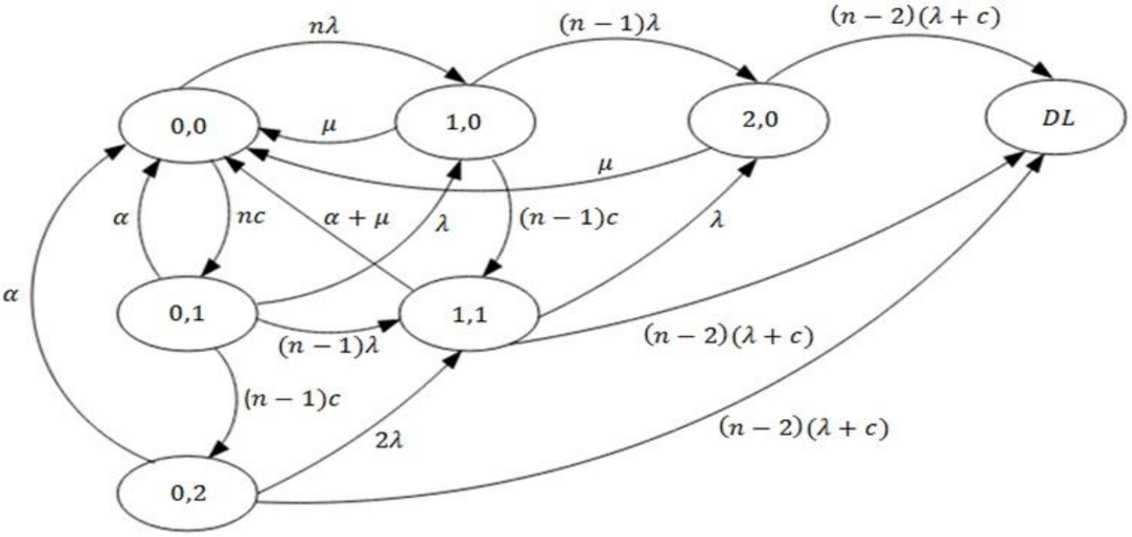

Figure-2. Intensity of transitions in the Markov chain for the case n-k=2.

Values of $Q_{l,m}$are calculated beginning from the states on the $l+m=n-k$ diagonal.

$$Q_{2,0} = \frac{1}{\mu+(n-2)(\lambda+c)}\left(\mu Q_{0,0} + (n-2)(\lambda+c)Q_{DL}\right);$$

$$Q_{2,0} \approx \frac{(n-2)(\lambda+c)}{\mu};$$

$$Q_{1,1} = \frac{1}{\alpha+\mu+\lambda+(n-2)(\lambda+c)}\left((\alpha+\mu)Q_{0,0} + \lambda Q_{2,0} + (n-2)(\lambda+c)Q_{DL}\right);$$

$$Q_{1,1} \approx \frac{(n-2)(\lambda+c)}{\alpha+\mu};$$

$$Q_{0,2} = \frac{1}{\alpha+2\lambda+(n-2)(\lambda+c)}\left(\alpha Q_{0,0} + 2\lambda Q_{1,1} + (n-2)(\lambda+c)Q_{DL}\right);$$

$$Q_{0,2} \approx \frac{(n-2)(\lambda+c)}{\alpha}.$$

Next, we examine the states on the $l+m=n-k-1$ diagonal:

$$Q_{1,0} = \frac{1}{\mu+(n-1)\lambda+(n-1)c}\left((n-1)\lambda Q_{2,0} + (n-1)cQ_{1,1} + \mu Q_{0,0}\right);$$

$$Q_{1,0} \approx \frac{n-1}{\mu}\left(\lambda\frac{(n-2)(\lambda+c)}{\mu} + c\frac{(n-2)(\lambda+c)}{\alpha+\mu}\right);$$

$$Q_{1,0} \approx \frac{(n-1)(n-2)(\lambda+c)}{\mu}\left(\frac{\lambda}{\mu} + \frac{c}{\alpha+\mu}\right);$$

$$Q_{0,1} = \frac{1}{\alpha+n\lambda+(n-1)c}\left(\lambda Q_{1,0} + (n-1)\lambda Q_{1,1} + (n-1)cQ_{0,2} + \alpha Q_{0,0}\right).$$

The value of $\lambda Q_{1,0}$is negligible in comparison with $\lambda Q_{1,1}$and $cQ_{0,2}$,so:

$$Q_{0,1} \approx \frac{1}{\alpha}\left((n-1)\lambda Q_{1,1} + (n-1)cQ_{0,2}\right);$$

$$Q_{0,1} \approx \frac{1}{\alpha}\left((n-1)\lambda\frac{(n-2)(\lambda+c)}{\alpha+\mu} + (n-1)c\frac{(n-2)(\lambda+c)}{\alpha}\right);$$

$$Q_{0,1} \approx \frac{(n-1)(n-2)(\lambda+c)}{\alpha}\left(\frac{\lambda}{\alpha+\mu} + \frac{c}{\alpha}\right).$$

The expected number of cycles till data loss is calculated in case of the initial (1,0) and (0,1) states:

$$n_{e,(1,0)} = \frac{1}{Q_{1,0}} \approx \frac{\mu}{(n-1)(n-2)(\lambda+c)}\left(\frac{\lambda}{\mu} + \frac{c}{\alpha+\mu}\right)^{-1};$$

$$n_{e,(0,1)} = \frac{1}{Q_{0,1}} \approx \frac{\alpha}{(n-1)(n-2)(\lambda+c)}\left(\frac{\lambda}{\alpha+\mu} + \frac{c}{\alpha}\right)^{-1}.$$

Mean cycle times for the initial (1,0) and (0,1) states are estimated. Values of $T_{l,m}$are calculated in the same sequence as the corresponding $Q_{l,m}$ values:

$$T_{2,0} = \frac{1}{\mu + (n-2)(\lambda + c)};$$

$$T_{2,0} \approx \frac{1}{\mu};$$

$$T_{1,1} = \frac{1}{\alpha + \mu + \lambda + (n-2)(\lambda + c)}\lambda T_{2,0} + \frac{1}{\alpha + \mu + \lambda + (n-2)(\lambda + c)};$$

$$T_{1,1} \approx \frac{1}{\alpha + \mu};$$

$$T_{0,2} = \frac{1}{\alpha + 2\lambda + (n-2)(\lambda + c)} \cdot 2\lambda T_{1,1} + \frac{1}{\alpha + 2\lambda + (n-2)(\lambda + c)};$$

$$T_{0,2} \approx \frac{1}{\alpha};$$

$$T_{1,0} = \frac{1}{\mu + (n-1)\lambda + (n-1)c}\left((n-1)\lambda T_{2,0} + (n-1)cT_{1,1}\right) + \frac{1}{\mu + (n-1)\lambda + (n-1)c};$$

$$T_{1,0} \approx \frac{1}{\mu};$$

$$T_{0,1} = \frac{1}{\alpha + n\lambda + (n-1)c}\left(\lambda T_{1,0} + (n-1)\lambda T_{1,1} + (n-1)cT_{0,2}\right) + \frac{1}{\alpha + n\lambda + (n-1)c};$$

$$T_{0,1} \approx \frac{1}{\alpha}.$$

For mean cycle times $t_{e,(1,0)}$ and $t_{e,(0,1)}$ for the $(1, 0)$ and$(0, 1)$ initial states the formulas take on form:

$$t_{e,(1,0)} = t_{0,0\to 1,0} + T_{1,0};$$
$$t_{e,(0,1)} = t_{0,0\to 0,1} + T_{0,1};$$

$$t_{e,(1,0)} \approx \frac{1}{n\lambda};$$

$$t_{e,(0,1)} \approx \frac{1}{nc}.$$

The storage MTTDL for this model is

$$MTTDL_{n-k=2} = \min\left(n_{e,(1,0)}t_{e,(1,0)}, n_{e,(0,1)}t_{e,(0,1)}\right) \approx min\left(\frac{\mu\left(\frac{\lambda}{\mu} + \frac{c}{\alpha+\mu}\right)^{-1}}{n\lambda(n-1)(n-2)(\lambda+c)}, \frac{\alpha\left(\frac{\lambda}{\alpha+\mu} + \frac{c}{\alpha}\right)^{-1}}{nc(n-1)(n-2)(\lambda+c)}\right).$$

Generalization of the extended model to arbitrary parameters (n, k)

Formulas obtained for the $n - k = 1$ and $n - k = 2$ special cases can be generalized for arbitrary parameters (n, k).

The scheme for the calculation of the $Q_{l,m}$ and $T_{l,m}$ values shown on Figure-3 assumes sequential calculation of values up to the $l + m = const$ diagonals beginning from the diagonal corresponding to the (l, m) states, for which $l + m = n - k$ when moving from top to bottom along the diagonals.

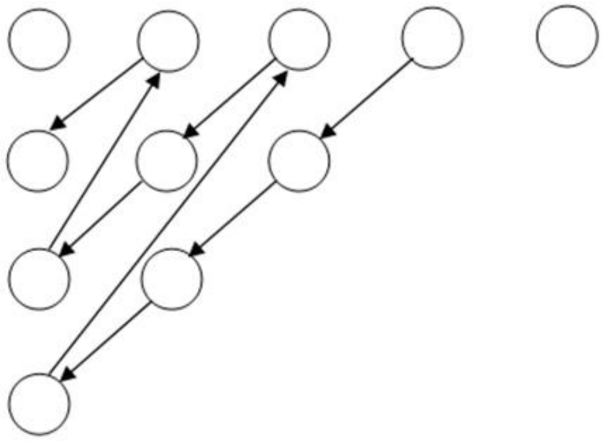

Figure-3. The scheme for calculating the values $Q_{l,m}$ and $T_{l,m}$.

The $t_{e,(1,0)}$ and$t_{e,(0,1)}$ mean cycle times for the model with (n,k) arbitrary parameters are calculated as follows.

$$T_{l,m}=\begin{cases}\dfrac{(n-l-m)\lambda T_{l+1,m}+(n-l-m)cT_{l,m+1}+1}{\mu+m\lambda+(n-l-m)(\lambda+c)}, l\neq 0, m=0,\\ \dfrac{m\lambda T_{l+1,m-1}+(n-l-m)\lambda T_{l+1,m}+(n-l-m)cT_{l,m+1}+1}{\alpha+\mu+m\lambda+(n-l-m)(\lambda+c)}, l\neq 0, m\neq 0,\\ \dfrac{m\lambda T_{l+1,m-1}+(n-l-m)\lambda T_{l+1,m}+1}{\alpha+m\lambda+(n-l-m)(\lambda+c)}, l=0, m\neq 0.\end{cases}$$

Omitting negligible elements, we can obtain:

$$T_{l,m}\approx\begin{cases}\dfrac{1}{\mu}, l\neq 0, m=0,\\ \dfrac{1}{\alpha+\mu}, l\neq 0, m\neq 0,\\ \dfrac{1}{\alpha}, l=0, m\neq 0.\end{cases}$$

The brought out formula suggests that the $T_{l,m}$ value depends only on the position of the (l,m) state within the $l+m=const$ diagonal, and does not depend on the diagonal.

So, the mean cycle times are:

$$t_{e,(1,0)}\approx\frac{1}{n\lambda};$$

$$t_{e,(0,1)}\approx\frac{1}{nc}.$$

Calculation of $Q_{l,m}$in arbitrary case is a bit more complex.

$$Q_{l,m}=\begin{cases}\dfrac{(n-l-m)\lambda Q_{l+1,m}+(n-l-m)cQ_{l,m+1}}{\mu+m\lambda+(n-l-m)(\lambda+c)}, l\neq 0, m=0,\\ \dfrac{m\lambda Q_{l+1,m-1}+(n-l-m)\lambda Q_{l+1,m}+(n-l-m)cQ_{l,m+1}}{\alpha+\mu+m\lambda+(n-l-m)(\lambda+c)}, l\neq 0, m\neq 0,\\ \dfrac{m\lambda Q_{l+1,m-1}+(n-l-m)\lambda Q_{l+1,m}+(n-l-m)cQ_{l,m+1}}{\alpha+m\lambda+(n-l-m)(\lambda+c)}, l=0, m\neq 0.\end{cases}$$

Bearing in mind that $\lambda Q_{l+1,m-1}$ negligible as compared to $\lambda Q_{l+1,m}$, and values of $cQ_{l,m+1}$and $Q_{l,m}$lie within the same $l+m=const$ diagonal and are comparable, while values of $Q_{l,m}$on inner diagonals are negligible as compared to values of $Q_{l,m}$on outer diagonals that are external in relation to these diagonals, the following conditions are true:

$$Q_{l_1,m_1}\ll Q_{l_2,m_2}, \text{where } l_1+m_1<l_2+m_2;$$

$$Q_{l,m}\approx\begin{cases}\dfrac{(n-l-m)\lambda Q_{l+1,m}+(n-l-m)cQ_{l,m+1}}{\mu}, l\neq 0, m=0,\\ \dfrac{(n-l-m)\lambda Q_{l+1,m}+(n-l-m)cQ_{l,m+1}}{\alpha+\mu}, l\neq 0, m\neq 0,\\ \dfrac{(n-l-m)\lambda Q_{l+1,m}+(n-l-m)cQ_{l,m+1}}{\alpha}, l=0, m\neq 0.\end{cases}$$

Let a "simple" path from the Q_{l_1,m_1} state into the Q_{l_2,m_2} state be called a sequence of transitions between states of the system that does not contain data recovery events as well as transitions between the states located on the same diagonal. Paths containing transitions between states on the same diagonal can be omitted since their probability as compared to "simple" paths is small.

It should be noted that the $Q_{l,m}$ value in the (l,m) state is defined as sum of terms S_i calculated along all "simple" paths from the $Q_{l,m}$ state to the Q_{DL} state. The number of such "simple" paths for the state $Q_{l,m}$is $2^{n-k-(l+m)}$, since the length of all "simple" paths from the $Q_{l,m}$ state into the Q_{DL} state is fixed and equal to $n-k-(l+m)$, and in each intermediate state there are two options for the further direction of the path – up or down.

The recurrent relations for $Q_{l,m}$ suggest that the S_i value corresponding to some fixed "simple" path is defined as product of transition intensity between states along this path divided by the product of data recovery intensities in each of the path states except the DL state.

Let's define $I(l,m)$ and $P(l,m)$ auxiliary functions:

$$I(l,m)=\begin{cases}\mu, l\neq 0, m=0,\\ \alpha+\mu, l\neq 0, m\neq 0,\\ \alpha, l=0, m\neq 0,\end{cases}$$

$$P(l,m) = \prod_{i=0}^{(n-k)-(l+m)} (n-(l+m)-i).$$

So, the $Q_{l,m}$ value for (l,m) arbitrary state is defined by the following expression:

$$Q_{l,m} \approx P(l,m)\left(\sum_{\beta_1,\dots,\beta_{(n-k)-(l+m)}=0}^{1} \frac{(\lambda+c)\prod_{i=1}^{(n-k)-(l+m)} \lambda^{\beta_i} c^{1-\beta_i}}{I(l,m)\prod_{i=1}^{(n-k)-(l+m)} I(l+\sum_{j=1}^{i}\beta_i\,, m+\sum_{j=1}^{i}(1-\beta_i))}\right);$$

$$Q_{l,m} \approx \frac{P(l,m)}{I(l,m)}\left(\sum_{\beta_1,\dots,\beta_{(n-k)-(l+m)}=0}^{1} \prod_{i=1}^{(n-k)-(l+m)} \frac{(\lambda+c)\lambda^{\beta_i} c^{1-\beta_i}}{I(l+\sum_{j=1}^{i}\beta_i\,, m+\sum_{j=1}^{i}(1-\beta_i))}\right).$$

The expected number of $n_{e,(1,0)}$ and $n_{e,(1,0)}$ cycles is defined by the following expressions

$$Q_{1,0} \approx \frac{P(1,0)}{I(1,0)}\left(\sum_{\beta_1,\dots,\beta_{(n-k)-1}=0}^{1} \prod_{i=1}^{(n-k)-1} \frac{(\lambda+c)\lambda^{\beta_i} c^{1-\beta_i}}{I(1+\sum_{j=1}^{i}\beta_i\,,\sum_{j=1}^{i}(1-\beta_i))}\right);$$

$$Q_{0,1} \approx \frac{P(0,1)}{I(0,1)}\left(\sum_{\beta_1,\dots,\beta_{(n-k)-1}=0}^{1} \prod_{i=1}^{(n-k)-1} \frac{(\lambda+c)\lambda^{\beta_i} c^{1-\beta_i}}{I(\sum_{j=1}^{i}\beta_i\,, 1+\sum_{j=1}^{i}(1-\beta_i))}\right);$$

$$n_{e,(1,0)} = \frac{1}{Q_{1,0}} \approx \left(\sum_{\beta_1,\dots,\beta_{(n-k)-1}=0}^{1} \frac{P(1,0)}{I(1,0)} \prod_{i=1}^{(n-k)-1} \frac{(\lambda+c)\lambda^{\beta_i} c^{1-\beta_i}}{I(1+\sum_{j=1}^{i}\beta_i\,,\sum_{j=1}^{i}(1-\beta_i))}\right)^{-1};$$

$$n_{e,(0,1)} = \frac{1}{Q_{0,1}} \approx \left(\sum_{\beta_1,\dots,\beta_{(n-k)-1}=0}^{1} \frac{P(0,1)}{I(0,1)} \prod_{i=1}^{(n-k)-1} \frac{(\lambda+c)\lambda^{\beta_i} c^{1-\beta_i}}{I(\sum_{j=1}^{i}\beta_i\,, 1+\sum_{j=1}^{i}(1-\beta_i))}\right)^{-1}.$$

Therefore, the analytical expression for MTTDL of the system for arbitrary (n,k) parameters:

$$MTTDL_{n,k} = \min\left(n_{e,(1,0)}t_{e,(1,0)}, n_{e,(0,1)}t_{e,(0,1)}\right) \approx$$

$$\approx \min\left(\begin{array}{c} \dfrac{1}{n\lambda(\lambda+c)}\left(\displaystyle\sum_{\beta_1,\dots,\beta_{(n-k)-1}=0}^{1} \frac{P(1,0)}{I(1,0)} \prod_{i=1}^{(n-k)-1} \frac{(\lambda+c)\lambda^{\beta_i} c^{1-\beta_i}}{I(1+\sum_{j=1}^{i}\beta_i\,,\sum_{j=1}^{i}(1-\beta_i))}\right)^{-1}, \\ \dfrac{1}{nc(\lambda+c)}\left(\displaystyle\sum_{\beta_1,\dots,\beta_{(n-k)-1}=0}^{1} \frac{P(0,1)}{I(0,1)} \prod_{i=1}^{(n-k)-1} \frac{(\lambda+c)\lambda^{\beta_i} c^{1-\beta_i}}{I(\sum_{j=1}^{i}\beta_i\,, 1+\sum_{j=1}^{i}(1-\beta_i))}\right)^{-1} \end{array}\right).$$

Results and comparative analysis of the classical and extended models

Comparison of the calculation results based on the proposed model with those obtained using the classical model is shown on Figure-4. In the proposed model, intensity of bit errors was defined by the mean intensity of disk access in the storage. As one can see on the chart, the results obtained in the classical model that account only for explicit disk failures differ from the refined results obtained within the proposed model. The observed difference is significant and confirms practical relevance of the proposed model. The obtained results are in line with the conclusions of[11] stating the need to account for latent errors and perform data checksum verification. The dependency of storage MTTDL on parameters (n,k) agrees quantitatively with results of [19], namely with behavior of conditional probability of data loss against parity fragments of erasure code.

The plots show that in this model the storage MTTDL largely depends on the mean disk usage intensity — with the decreasing of disk usage intensity the bit error rate also decreases and storage MTTDL becomes closer to the MTTDL values computed using a simpler model that doesn't account for the irrecoverable bit errors.

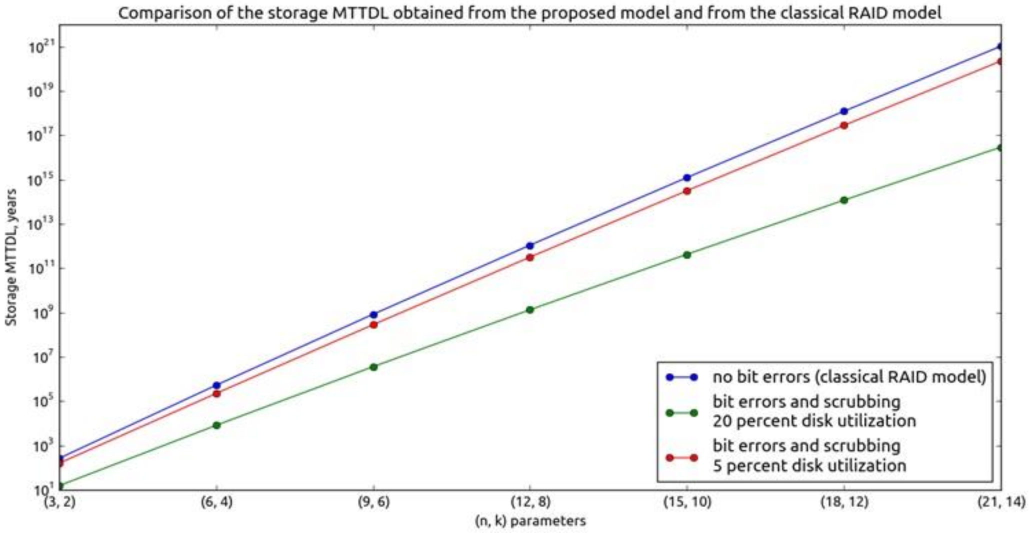

Figure-4. Comparison of results obtained in the model that accounts for irrecoverable bit errors and scrubbing process and those from the simpler model.

The graph of data storage MTTDL dependence on the mean scrubbing interval shows that on the initial section, MTTDL is constant and virtually does not depend on scrubbing intensity. Then, it starts to go down abruptly as the mean scrubbing interval increases (see. Figure-5). This happens because the formula for the storage MTTDL contains two expressions, from which the minimum one is selected. At the point where MTTDL starts to decrease the expression with the minimum value changes.

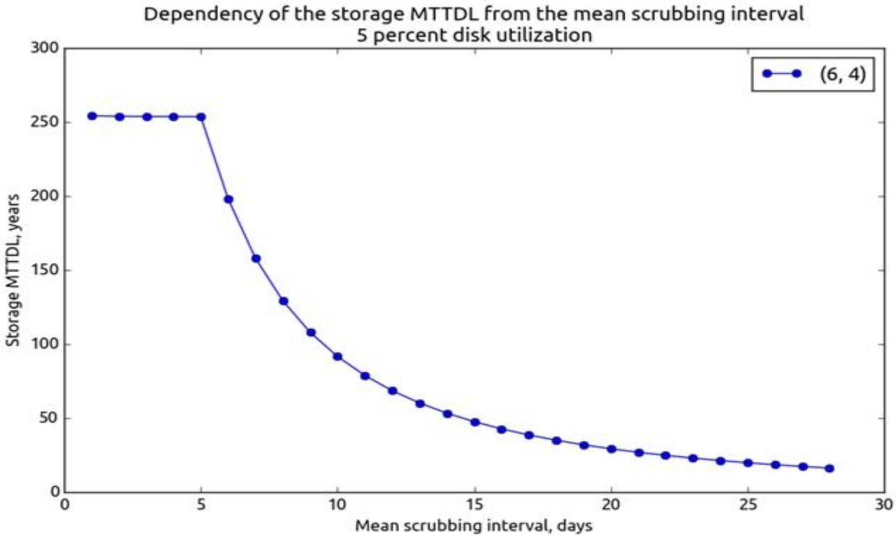

Figure-5. Dependence of the data storage MTTDL on the mean scrubbing interval.

The graph (see Figure-5) shows that the proposed method allows us to find the balance between data storage reliability and financial costs imposed by the continuous scrubbing process.

5. DISCUSSION

The proposed model offers more flexible and reliable scheme than those describing RAID-5 and RAID-6 data redundancy schemes. The scrubbing of RAID-groups involves the complete scanning of the whole disks, leading to prolonged timeframes of reduced performance and reliability.

The study [15] considers the similar problem of improving reliability of RAID-group storages by means of disk scrubbing. Despite the differences in RAID and error coding schemes, the results from corresponding models are in agreement with each other. The plot (Figure-6) demonstrates MTTDL dependence on scrubbing period for an event-driven simulated 10 PB system of RAID-6 groups with scrubbing IO-load at ten per cent. The set of underlying parameters represent SATA drives with capacity of 300 GB, probabilities of irrecoverable bit and sector errors of 〖10〗^(-14) and 4.096×〖10〗^(-11)correspondingly.

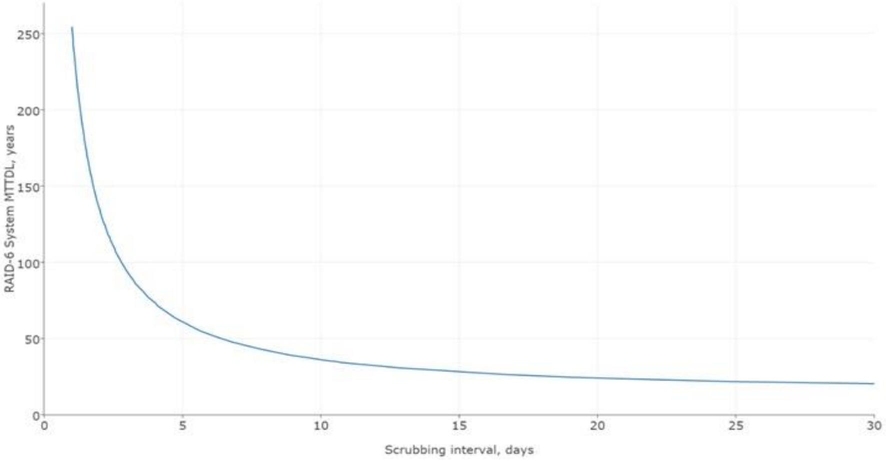

Figure-6. Dependence of the RAID-6 data storage MTTDL on the scrubbing interval [15].

The results of [15] show that there is a lower bound for a scrubbing interval imposed by the system workload saturation. The new model proposed above already considers this factor and represents it as the initial plateau on a respective storage MTTDL curve.

6. CONCLUSIONS

This work presents a new math model of distributed storage reliability with an analytical description based on the two-dimensional Markov chains. A new method of the storage MTTDL calculation using approximate analytical expressions for the arbitrary (n,k)encoding parameters is proposed. The work demonstrates that the lack of the consideration ofthe bit read errors in classical disk array reliability models results in overstated MTTDL estimations. Therefore, latent faults substantially affect reliability of the data storage, and this fact should not be disregarded. Using obtained analytical expression for the MTTDL, the work shows that the proposed method allows finding the optimal ratio of data storage reliability and financial expenses imposed by the continuous scrubbing process.

In the discussion section, we have compared the results of the proposed approach to reliability estimation with ones presented for the petabyte systems based on RAID-groups. The observed quantitative similarity serves as an additional validation proof of the considered analytical model.

7. FURTHER STUDY

The investigated methods of improving reliability of the super large data storage allow advancements on the following perspective research topics. One of the significant problems is the estimation of the checksum validation overhead followed by the optimization of the data validation cost under the given storage reliability requirements. The related problem is to construct a validation algorithm for stored data blocks and system components with different access patterns. As a first potential solution, one can increase the data check rate by skipping data frequently accessed by the clients. The second solution lies in reducing the overhead of waking disks up from the stand-by power saving state. One can achieve this by postponing the checksum validation for a fragment until a read access occurs for a disk that contains this fragment. Generally, it is plausible to group validation requests for the data on same disks. In [20]Schwarz*et al.* support the similar idea of opportunistic scrubbing. The second problem is an investigation of disk failure prediction methods. These predictions can rely on the SMART-statistics analysis based on the different machine learning algorithms. The study can include estimation for algorithm prediction accuracy and optimization of the prediction algorithms to increase their accuracy and performance as well as their integration into the data storage reliability model.

ACKNOWLEDGEMENTS

The article is published within applied scientific research performed with financial support of the Ministry of Education and Science of the Russian Federation. Subsidy provision agreement 14.579.21.0010. Universal identifier of the agreement is RFMEFI57914X0010.

REFERENCES

[1] Patterson D. A., GibsonG. and Katz R. H.1988. A Case for Redundant Arrays of Inexpensive Disks (RAID). Proc. of ACM SIGMOD, pp. 109-116.

[2] Reibman A. and TrivediK. S. 1989.Transient Analysis of Cumulative Measures of Markov Model Behavior. Communications in Statistics-Stochastic Models, 5:683–710.

[3] Schultz M., GibsonG., KatzR., and PattersonD.1989. How Reliable is a RAID? Proceedings of Comp. Con, pp.118 -123.

[4] Malhotra M. and TrivediK. S.1993. Reliability Analysis of Redundant Arrays of Inexpensive Disks. Journal of Parallel and Distributed Computing, Special issue on parallel I/O systems. 17(I.1-2), pp.146-151.

[5] Greenan K. M., Plank J. S. and WylieJ. J.2010. Mean Time to Meaningless: MTTDL, Markov models, and Storage System Reliability. Proceedings of the 2nd USENIX conference on Hot topics in storage and file systems, pp.1-5.

[6] KarmakarP.andGopinathK.2015.Are Markov Models Effective for Storage Reliability Modelling? arXiv:1503.07931v1.

[7] LiY., Lee P. P. and LuiJ.2013. Stochastic analysis on raid reliability for solid-state drives. IEEE 32nd International Symposium on Reliable Distributed Systems (SRDS). IEEE, pp. 71-80. (http://arxiv.org/pdf/1304.1863.pdf).

[8] Mann S. E., Anderson M. and RychlikM.2012.On the Reliability of RAID Systems: An Argument for More Check Drives. arXiv:1202.4423v1.

[9] Pâris J.-F., Schwarz T. J., Amer A., and Long D. D. E.2010.Improving Disk Array Reliability Through Expedited Scrubbing. Proceedings of the 5th IEEE International Conference on Networking, Architecture, and Storage, pp. 119-125.

[10] Xin Q., Miller E. L., Schwarz T. J., Long D. D. E., Brandt S. A. and LitwinW.2003.Reliability mechanisms for very large storage systems. Proceedings of the 20th IEEE/11th NASA Goddard Conference on Mass Storage Systems and Technologies, pp. 146–156.

[11] Elerath J. G. and PechtM.2007. Enhanced Reliability Modeling of RAID Storage Systems. 37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, pp. 175-184.

[12] Ivanichkina L. and NeporadaA.2014.Mathematical methods and models of improving data storage reliability including those based on finite field theory. Contemporary Engineering Sciences, 7 (28), pp. 1589-1602 http://dx.doi.org/10.12988/ces.2014.411236.

[13] Douceur J. and BoloskyW.1999.A Large-Scale Study of File-System Contents, ACM SIGMETRICS Performance Review, pp. 59-70.

[14] Huang H., Hung W., and Shin K. G.2005.FS2: Dynamic Data Replication in Free Disk Space for Improving Disk Performance and Energy Consumption, 18th ACM SIGOPS Operating Systems Review, 39 (5), pp. 263-276.

[15] Iliadis I., Haas R., Hu X., and EleftheriouE.2008.Disk Scrubbing Versus Intra-Disk Redundancy for High-Reliability RAID storage System. In: Proceedings of the 2008 ACM SIGMETRICS international conference on Measurement and modeling of computer systems, 36 (1), pp. 241-252.

[16] Liu J., Zhou K., Wang Z., Pang L.and Feng D. 2010. Modeling the Impact of Disk Scrubbing on Storage System. Journal of Computers, 5(11), pp. 1629-1637.

[17] Venkatesan V. and Iliadis I. 2013. Effect of latent errors on the reliability of data storage systems.IEEE21st International Symposium on Modeling, Analysis & Simulation of Computer and Telecommunication Systems (MASCOTS), pp. 293-297.

[18] Tamo I., Papailiopoulos D. S. and Dimakis A. G. 2013. Optimal locally repairable codes and connections to matroid theory. Proceedings of IEEE International Symposium on Information Theory (ISIT),pp. 1814-1818.

[19] Li X., Lillibridge M. and Uysal M. 2011. Reliability analysis of deduplicated and erasure-coded storage.ACM SIGMETRICS Performance Evaluation Review, 38(3), 4-9.

[20] Schwarz T. J., Xin Q., Miller E. L., Long D. D., Hospodor A. and Ng S.2004. Disk scrubbing in large archival storage systems. (MASCOTS 2004). Proceedings. The IEEE Computer Society's 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems, pp. 409-418.