Besson Mathieu

Tanguy Christian
Orange Labs, Issy-les-Moulineaux, France

Grall Antoine
LM2S, Troyes, France

On the average file unavailability for specific storage disk arrangements in Cloud systems

Keywords

cloud, storage system design, *k*-out-of-*n* systems

Abstract

Cloud computing is a growing field since data storage is becoming ever more decentralized. Providers of Cloud solutions want to insure the safety and availability of their customers' data. In order to increase these performance indices, several storage policies have been implemented: replication, erasure codes, etc. A few of them rely on randomized procedures.
In this paper, we focus on the influence of a specific storage policy on the availability of a given file. Taking only disk failures into account, we provide a general formula for the average file unavailability $\overline{U}$, which is a generalization of the well-known *k*-out-of-*n* problem, to which it reduces when disks are identical. We then calculate $\overline{U}$ for several configurations when disks have different reliabilities, and show that the disk arrangement has a major impact on the result. We also provide an approximation which could be helpful for more complex arrangements.

1. Introduction

Cloud services have emerged as the new data storage solution. Business players are competing to offer the best solution in terms of usability, safety and availability [2]. The key point is to design robust data centers that maximize these performance criteria.
From a customer point of view, data availability is crucial. That is why many storage solutions have been developed in order to increase data safety. They use pure replication (several copies of the same data are stored in different locations) [9], [12]-[13], [19], [24], compression [1], or erasure code techniques (based on error-correcting codes that lessen the storage overhead) [7]-[8], [23], etc.
How can we compare the performance of different approaches/architectures? Several performance indices have been proposed:

- the Mean Time To Data Loss (MTTDL) is the most commonly used performance index [7], [9], [15], [23]-[24] in the context of data loss prevention strategies. Different models of storage and repair policies have been studied, and "rebuild'' procedures described [22]. The MTTDL varies with the way redundant data are stored ("Clustered vs Declustered'') [24]. The MTTDL is often computed using Markov models and criticized accordingly [5]-[6], [16].
- the Normalized Magnitude of Data Loss (NOMDL) [6] is expressed by the number of bytes lost per mission lifetime.
- the Expected Annual Fraction of Data Loss (EAFDL) [9].
- other metrics have been suggested: the Bit Half-life [16], the Double Disk Failures [5], the Data Loss events per Petabyte Year [7] and others, showing that the definition of a metric combining ease of computation, meaningfulness, and practicality is still hotly debated.

In this work, we have adopted the point of view of a customer, and chosen to consider the average unavailability $\overline{U}$ of a given file, in the case of a specific storage policy [12]-[13], [21], [23]-[24].

While this unavailability is of course dependent on the architecture of the data center in terms of hardware (disks, buses, switches, racks, nodes, etc.) and software (protocols), we shall limit ourselves to the influence of the hard drives' unavailabilities only. Our configuration is actually a generalization of the well-known *k*-out-of-*n* problem [3], [11], [16]. We show that the disk placement policy has a direct influence on the file unavailability when the hard drives are different. We also provide a satisfactory approximation to $\overline{U}$ in order to make computations of this performance index much easier.

Our paper is organized as follows. In section 2, we describe the storage policy. In section 3, we explain the method leading to the general expression of the file unavailability and explain how it reduces to the classical *k*-out-of-*n* result when disks are identical. Because of combinatorial aspects, the exact expression of the file unavailability cannot always be computed in a reasonable time. We explicitly calculate $\overline{U}$ in section 4 for particular cases that could be deployed in real systems. From the exact expressions, we deduce efficient second-order approximations that provide quick and satisfactory estimates that might be helpful in the general case. Finally, by comparing the results of our different case studies, we prove the influence of the disk arrangement on the file unavailability. We conclude by a brief discussion of future work.

2. Storage Policy

2.1. Data processing

Each file is first split into K data blocks. Redundancy procedures implemented in erasure coded systems transform those K blocks into n new blocks (also called chunks) which are then stored in different locations, in order to minimize common-cause failures. Let us set

$$d = n - K + 1. \tag{1}$$

The gist of such procedures is that the initial file cannot be rebuilt (i.e., recovered) if at least d chunks have been lost because of hard drive failures (d may be linked to some Hamming distance [7]). In other words, if at least d of the n chunks are lost, the file is irrecoverable. In practice, we shall have $7 \leq n \leq 20$ and $3 \leq d \leq 6$.

2.2. Data storage

We now have to store n chunks in the set of m hard drives of the system. The storage policy is implemented using randomized procedures, the principle of which is displayed in *Figure 1*. Firstly, we randomly select one disk (among the m disks) where we will store the first chunk. Secondly, the $n - 1$ remaining chunks are stored (again, at random) in the S disks following the first one. The disks are indexed by a logical address; they are not necessarily located in the same rack or node. S is called the spread factor. It must obey the inequalities $n - 1 \leq S \leq m - 1$. The number of disks in a data center can reach several thousands. In the following, the set of n disks containing the chunks will be called a configuration. If the spread factor S is large enough, there might be different ways of selecting the same configuration C, as shown in *Figure 2*. For operational reasons, we do not wish one configuration to be "drawn" more often than others, since it would imply a heavier load on the relevant disks, and a possible decrease of their lifetimes. We can ensure that the load is evenly distributed over all possible configurations by choosing

$$S < \frac{m}{2}, \tag{2}$$

because any possible configuration can then be selected once only.

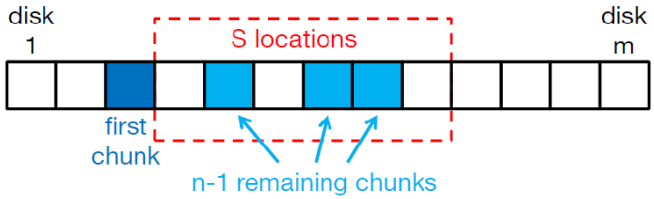

Figure 1. The placement policy using the spread factor $S = 6$ for $m = 13$ and $n = 4$

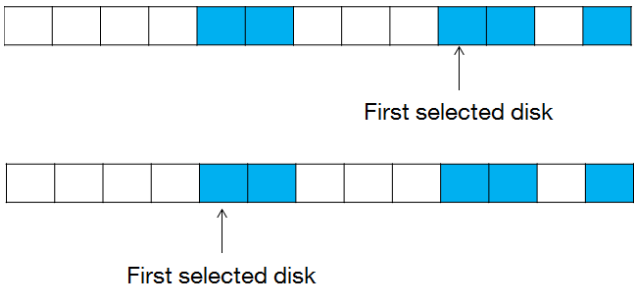

Figure 2. An example of two different ways to select a configuration with $m = 13$, $S = 10$ and $n = 5$

The probability of selecting a configuration is:

$$\frac{1}{m\, C_S^{n-1}} \tag{3}$$

where

$$C_S^{n-1} = \frac{S!}{(S-n+1)!\,(n-1)!}. \quad (4)$$

In this work, we wish to study the possible influence of S on the file unavailability, so that it will be treated as a (almost) free parameter, as will be apparent in the next sections.

3. File unavailability

3.1. General case

For a system of disks with unavailabilities $\{q_i\}_{1\le i\le m}$, the file unavailability can be calculated as follows. For any configuration C of n disks, the probability that at least d chunks are lost is known as "d-out-of-n:F" [3], [11], [16], [21]. Let us name this quantity $U(d\ \text{out}-\text{of}-n\text{:}\ F\ ;C)$. Efficient methods and algorithms have been proposed to compute it, using various formulations [4], [11]. Then, the average file unavailability is nothing but the sum over all feasible configurations of $U(d\ \text{out}-\text{of}-n\text{:}F\ ;C$ multiplied by the probability that C has been selected. Under the assumption $m > 2\,S$, all configurations have the same probability of being drawn, so that the average file unavailability reads

$$\bar{U} = \frac{1}{m\ C_S^{n-1}} \sum_C U(d\ \text{out}-\text{of}-n\text{:}\ F;C)\,. \quad (5)$$

It is worthwhile stressing that $\bar{U}$ is actually a two-fold average. The first average is related to the randomized procedure for the storage of chunks. The second one is implicit in $U(d\ \text{out}-\text{of}-n\text{:}\ F;C)$ since it must take the random character of failures and repairs of each disk into account, so that individual unavailabilities may be defined. Equation (5) shows that $\bar{U}$ can be computed exactly in the general case, but the C_S^{n-1} factor makes computation times unreasonably long when $S \gg n-1$. We shall see in the following that eq. (5) may however take a simple form in particular cases.

3.2. Identical disks

When disks are identical, with therefore the same reliabilities, $U(d\ \text{out}-\text{of}-n\text{:}\ F\ ;C)$ reduces to the classical result for k-out-of-n systems. Indeed, if q is the disk unavailability, the probability that at least d chunks are lost for any configuration C is

$$U(q) = \sum_{k=d}^{n} C_n^k\, q^k (1-\ q)^{n-k}. \quad (6)$$

By replacing the previous expression in eq. (5), the average file unavailability for identical disks reads

$$\bar{U} = U(q). \quad (7)$$

3.3. Discussion

The lifetime duration of disks in data centers is shorter than the one of disks in personal computers since disks in Cloud systems are much more solicited. The issue of the disk's MTTF is still much debated [5], [10], [14]-[16], [18], and the figures mentioned in the literature are typically of the order of a few 10^5 hours. A good order of magnitude for a disk unavailability would thus be $q \approx 10^{-4}$ or even less. For such values, even the first term of eq. (7), namely $C_n^d q^d (1-q)^{n-d}$, that can also be approximated by $C_n^d q^d$, provides a good estimate of the file unavailability. Indeed, with $n = 12$ and $d = 4$, the relative error between eq. (7) and its approximation is less than 0.07 %.
Equation (7) and its approximation give the file unavailability when disks are identical. However, it is known that even for batches of the same model, different lifetimes are to be expected [15]. In the following section, we shall study systems in which two or more families of disks are used, in different deployments.

4. Case studies

In this section we shall consider two different ways to arrange disks, and see their possible influence on $\bar{U}$. We shall also inquire whether a good estimate of $\bar{U}$ could be obtained from $U(\bar{q})$, where $\bar{q}$ is the average disk unavailability.

4.1. Block arrangement

Let us first consider a system constituted by two types of disks, of unavailabilities q_1 and q_2, respectively. The first deployment considered here is represented in *Figure 4*, in which m_1 disks of type 1 are followed by m_2 disks of type 2. This placement will be denoted as "two-block" arrangement.

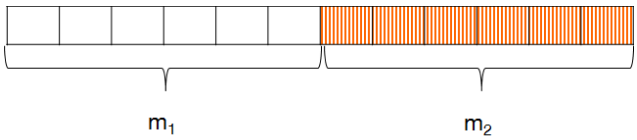

Figure 4. "Two-block" arrangement with m = 12 and $m_1 = m_2 = 6$

The numbers m_1 and m_2 can be arbitrary, but we will assume that both of them are greater than S. For the

sake of simplicity, let us start the calculation of $\overline{U}$ for the simplest case, that is $m_1 = m_2 = m/2$, and set

$$q = \frac{1}{2}(q_1 + q_2), \tag{8}$$

$$\eta = \frac{1}{2}(q_1 - q_2). \tag{9}$$

$\overline{U}$ is the sum of four contributions

$$\overline{U} = V_1 + V_2 + V_{1\to 2} + V_{2\to 1}. \tag{10}$$

The term V_1 (resp. V_2) originates with the selection of the first disk in the first $m_1 - S$ disks (resp. $m_2 - S$) of the system. Indeed, if we do so, the S following neighbors will also be of type 1. Then, any configuration from these S+1 disks will be constituted with type-1 disks only (see the top of *Figure 5*) and we can apply eq. (7).

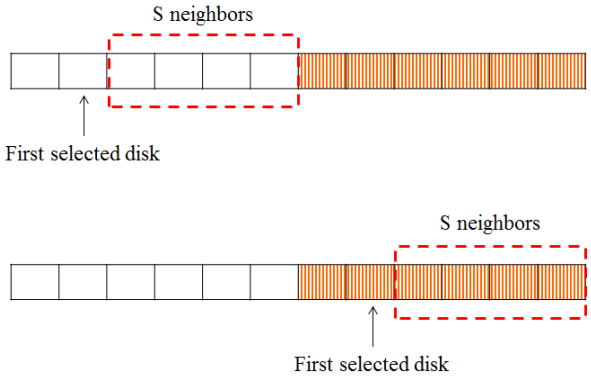

Figure 5. Contribution of configurations with the same type of disks with $m = 12$ and $S = 4$

For all these configurations, the probability of file unavailability will be, thanks to eq. (2),

$$V_1 = \frac{m_1 - S}{m} U(q_1). \tag{11}$$

We can make a Taylor expansion in η of $U(q_1)$, with $q_1 = q + \eta$. Since $U(q)$ is of degree n, the derivatives $U^{(k)}(q)$ with $k > n$ vanish. V_1 reads therefore

$$V_1 = \frac{m_1 - S}{m} \sum_{k=0}^{n} \frac{1}{k!} U^{(k)}(q)\, \eta^k. \tag{12}$$

In a similar way (see the bottom of *Figure 5*), if the first selected disk belongs to the first $m_2 - S$, the contribution V_2 reads, because $q_2 = q - \eta$,

$$V_2 = \frac{m_2 - S}{m} \sum_{k=0}^{n} \frac{(-1)^k}{k!} U^{(k)}(q)\, \eta^k. \tag{13}$$

The remaining terms are obtained when the S+1 disks could contain disks of both types. The contribution $V_{1\to 2}$ (resp. $V_{2\to 1}$) is obtained when the first disk selected belongs to the last S disks of type 1 (resp. type 2) (see *Figure 6*).

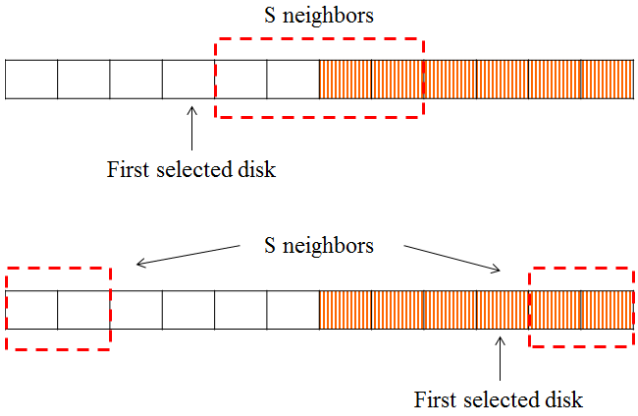

Figure 6. The contributions when the $\boldsymbol{S+1}$ disks are of both types

$V_{1\to 2}$ and $V_{2\to 1}$ are expected to be expressed, like V_1 and V_2 in eqs. (12) and (13), as linear combinations of derivatives of $U(q)$ and powers of η. Indeed, for any of these configurations, the probability of losing at least d disks is a polynomial expression in q_1 and q_2, as is the sum of all contributions. $V_{1\to 2}$ will be therefore a polynomial in q and η of degree at most n. The only unknown is the coefficient before each term. These coefficients are expected to depend on (at most) n, S, d and m. We have computed formally $V_{1\to 2}$ and these coefficients for various values of n, d and S, using Mathematica. We have found that

$$V_{1\to 2} = \frac{1}{m} \sum_{k=0}^{n} \alpha(k)\, U^{(k)}(q)\, \eta^k, \tag{14}$$

where

$$\alpha(k) = \begin{cases} \frac{S-n+1}{n\,k!}, & (k \text{ odd}) \\ \frac{(n-k)\,S-k\,(n+1)}{n\,(k+1)!}, & (k \text{ even}) \end{cases} \tag{15}$$

We see in eq. (14) that the contribution of d is implicit and only comes from the derivatives of $U(q)$ (see eq. (6)). The "symmetric" contribution $V_{2\to 1}$ is obtained by merely replacing η by $-\eta$ in eq. (14).
We deduce from eqs. (12-15) and $\alpha(0) = S$ that

$$\bar{U} = U(q) + \left(1 - \frac{2\,S}{m}\right) \sum_{\substack{k=2 \\ k \text{ even}}}^{n} \frac{1}{k!}\, U^{(k)}(q)\, \eta^k + \frac{2}{m} \sum_{\substack{k=2 \\ k \text{ even}}}^{n} \alpha(k)\, U^{(k)}(q)\, \eta^k \qquad (16)$$

Using the expression of $V_{1\to 2}$ we can generalize the expression of $\bar{U}$ for any m_1 and m_2, or more generally for a family of k types of disks ordered by blocks. Let us note $q_i = \bar{q} + \delta_i$ the unavailability of disks of type i, where

$$\bar{q} = \frac{1}{m} \sum_{i=1}^{k} m_i\ q_i \qquad (17)$$

is the average disk unavailability. We still call $V_{i\to j}$ the contribution when the first selected disk is of type i and the following block is of type j. Consequently,

$$V_{i\to j} = \frac{S}{m} U\left(\bar{q} + \frac{\delta_i + \delta_j}{2}\right) + \frac{1}{m} \sum_{k=1}^{n} \alpha(k)\, U^{(k)}\left(\bar{q} + \frac{\delta_i + \delta_j}{2}\right)\left(\frac{\delta_i - \delta_j}{2}\right)^k . \qquad (18)$$

Finally,

$$\bar{U}(q_1, \ldots, q_k) = \sum_{i=1}^{k} \frac{m_i - S}{m} U(q_i) + V_{1\to 2} + \cdots + V_{k\to 1}\ . \qquad (19)$$

Note that all the m_i are assumed to be greater than S for eq. (19) to be valid; this implies $S < m/k$. To obtain a quick and good approximation of eq. (19), we can perform a second-order expansion in all the δ_i's. After some work, we obtain

$$\bar{U}\left(q_1, \ldots, q_k\right) \approx U(\bar{q}) + \frac{1}{2} U^{(2)}(\bar{q})\, [(\Delta q)^2 - Q^2] \qquad (20)$$

where

$$(\Delta q)^2 = \frac{1}{m} \sum_{i=1}^{k} m_i (q_i - \bar{q})^2 \qquad (21)$$

is the variance of the distribution $\{q_i\}$, and

$$Q^2 = \frac{S+1}{6\,m} \frac{n+1}{n} \sum_{j \text{ follows } i} \left(q_i - q_j\right)^2 . \qquad (22)$$

We see that the file unavailability is equal to $U(\bar{q})$ plus corrections, the leading term of which is proportional to $U^{(2)}(\bar{q})$. The associated prefactor is the sum of the variance (always positive) and of a negative contribution including a $(S + 1)$ term characteristic of the transition zone (i.e., when $S + 1$ consecutive disks are of two types). While the two contributions are of opposite signs, we expect the overall sign to be positive, and therefore $\bar{U} > U(\bar{q})$, because of the constraint $S < m/k$ and typical values of n. By keeping S fixed and increasing m, we would obtain the same result. When $m_1 = m_2$, the second-order approximation gives

$$\bar{U} - U(q) \simeq \left(1 - \frac{4(S+1)(n+1)}{3\,m\,n}\right) \frac{U''(q)}{2} \eta^2 \qquad (23)$$

As indicated above, we restricted ourselves to the case $m > 2\,S$, so that $\bar{U} > U(q)$. The unavailability of the file in the two-block arrangement is greater than it would be for a homogeneous set of disks, of unavailability q.

4.2. Alternate arrangement

There is another simple way to deploy disks of unavailabilities q_1 and q_2, namely the alternate placement, represented in *Figure 7*.

Figure 7. The alternate arrangement with $\boldsymbol{m = 12}$, $\boldsymbol{m_1 = m_2 = 6}$

Obviously, for this kind of architecture, we must have $m_1 = m_2$. If we consider the $S + 1$ disks selected after the first step, we only have two different patterns, depending on the first selected disk (see *Figure 8*).

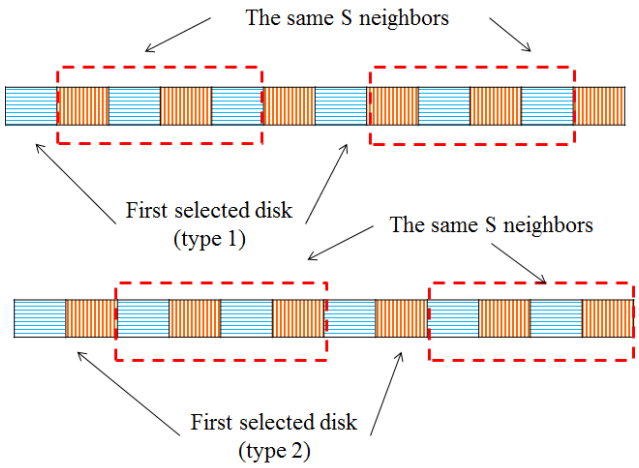

Figure 8. The pattern of the *S* neighbors when the first selected disk if of type 1 or 2

The only parameter that affects the form of the set of $S+1$ disks is the spread factor. If it is even, and if the first disk selected is of type 1, we count $S/2+1$ disks of type 1 and $S/2$ disks of type 2 while if the spread factor is odd we have $(S+1)/2$ disks of both types (*see Figure 9*).

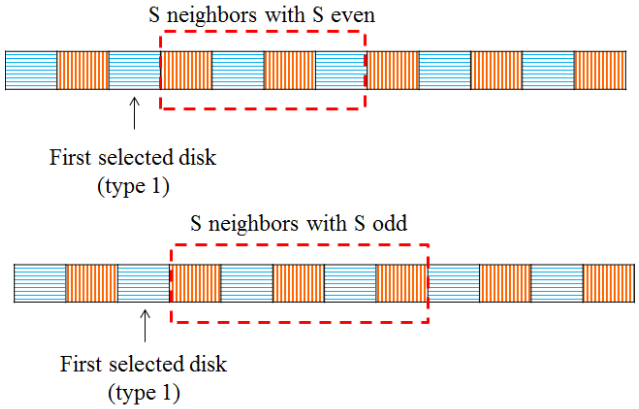

Figure 9. The number of disks of type 1 and 2 in the *S* neighbors of the first selected disk as a function of the parity of the spread factor

If the first disk selected is of type 2, the structure is the same as above (interchange type 1 and type 2). $\bar{U}$ is thus the sum of two contributions:

$$\bar{U} = \frac{1}{2}(W_1 + W_2), \qquad (24)$$

where W_1 is the contribution of the $\frac{m}{2} C_S^{n-1}$ configurations when the first selected disk is of type 1; likewise, W_2 is the relevant contribution when the first disk is of type 2. As in section **4.1** devoted to the block arrangement, we have computed (for different values of *d*, *n* and *S*) the formal expression of $\bar{U}$ as a function of q_1 and q_2, and therefore of q and η given again by eqs. (8) and (9). We have been able to identify the exact expression

$$\bar{U} = \sum_{p=0}^{\left\lfloor \frac{n}{2} \right\rfloor} (-\eta^2)^p \frac{U^{(2p)}(q)}{(2\,p)!} \frac{C_{\left\lfloor \frac{S}{2} \right\rfloor}^{p}}{C_S^{2p}} \begin{cases} \frac{n-2p}{n} & (S \text{ even}) \\ 1 & (S \text{ odd}) \end{cases} \qquad (25)$$

where $\lfloor x \rfloor$ is the integer part of *x*. As previously, we can restrict ourselves to a second-order approximation:

$$\bar{U} \approx U(q) - \frac{U^{(2)}(q)\eta^2}{2} \begin{cases} \frac{1}{S} & (S \text{ odd}) \\ \frac{n-2}{n(S-1)} & (S \text{ even}). \end{cases} \qquad (26)$$

In contrast to eq. (23), the second-order correction is always negative, and thus $\bar{U} < U(q)$. The alternate arrangement is performing better than the block arrangement.

We have also considered more than two families of disks. For three families of disks arranged as $q_1\ q_2\ q_3\ q_1\ q_2\ q_3$, etc. we have not yet found a general formula such as eq. (25). However, we have been able to find, with $\bar{q} = \frac{q_1+q_2+q_3}{3}$ and $(\Delta q)^2 = \frac{{\delta_1}^2+{\delta_2}^2+{\delta_3}^2}{3}$,

$$\bar{U} \approx U(\bar{q}) - \frac{U^{(2)}(\bar{q})\,(\Delta q)^2}{2} \begin{cases} \frac{1}{S} & (S = 3\,r+2) \\ \frac{n-1}{n\,S} & (S = 3\,r+1) \\ \frac{n-2}{n(S-1)} & (S = 3\,r) \end{cases} \qquad (27)$$

In the case of four families or more, even for the second-order corrections, the expressions become more complicated: there are contributions from terms other than $(\Delta q)^2$, showing again that the placement of disks has an influence on $\bar{U}$.

4.3. Discussion

In the previous subsections, we have studied two specific disk arrangements, for which we have obtained the exact file unavailability $\bar{U}$, for arbitrary values of *d*, *n*, *m*, *S*, *q* and η. These exact expressions may be computed very quickly. If we only take disk failures into account, the alternate arrangement gives better results in terms of file unavailability. $\bar{U}$ varies with the arrangement of disks of unequal

unavailabilities, as demonstrated by eqs. (16), (19), (25), and *Figure 10*.
In more complex arrangements, the exact calculation might be not so easy, if not hopeless. For this reason, we wish to provide reasonably accurate estimates of $\bar{U}$, which might prove sufficient for the design and assessment of data center architectures:

- a first simple estimate is $U(\bar{q})$, to which $\bar{U}$ reduces when all the unavailabilities q_i are equal, of course.
- we expect corrections to include the successive derivatives $U^{(l)}(\bar{q})$, multiplied by prefactors depending on all the $\delta_i = q_i - \bar{q}$. Note that the term with l = 1 necessarily cancels because of the definition of $\bar{q}$. Decent approximations could be found by keeping only the second-order term in $U^{(2)}(\bar{q})$.

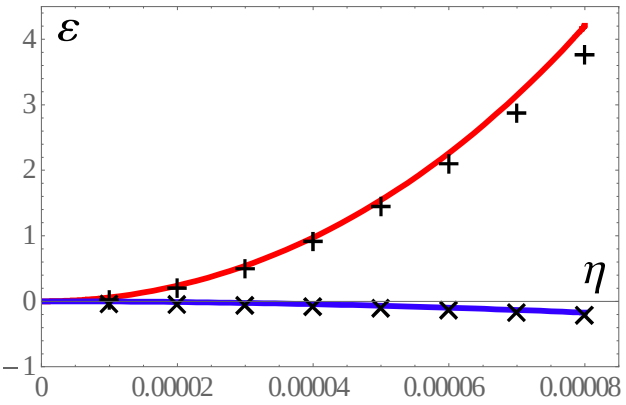

Figure 10. Exact values of $\boldsymbol{\varepsilon = \frac{\bar{U}}{U(\bar{q})} - 1}$ for d = 4, n = 15, S = 20, m = 3000, and $\boldsymbol{q = 10^{-4}}$ as functions of $\boldsymbol{\eta}$: block arrangement (full red line), alternate arrangement (full blue line), also represented by '+' and '×' are their respective second-order approximations given in eqs. (23) and (26)

In order to justify this approach, we have plotted (see *Figure 10*) as a function of η the quantity

$$\varepsilon = \frac{\bar{U}}{U(\bar{q})} - 1 \,, \qquad (28)$$

which gives a good indication of the relative error made by replacing $\bar{U}$ by $U(\bar{q})$. We have also represented by '+' and '×' the values obtained when replacing $\bar{U}$ by its second-order approximations. Clearly, only when η is large do these approximations differ from the exact results, especially in the block arrangement case; they are still satisfactory even when ε is a few hundred percents. Consequently, we recommend to use second-order approximations to $\bar{U}$, which will be easy to compute and yet accurate enough.

5. Conclusion

In a context of intense competition in terms of usability, safety, and availability of Cloud services, we have modelled the file unavailability for a specific erasure-coding storage policy. This model amounts to a generalization of the well-known *k*-out-of-*n* problem, to which it reduces when all the disks are identical.
In the real world, however, they have different reliabilities. We have studied several disk arrangements for which the average file unavailability $\bar{U}$ has been calculated exactly, and shown that these arrangements do matter. Our results indicate that while in the general case the numerical computation of $\bar{U}$ may be very cumbersome or even downright impossible in a reasonable amount of time, simple second-order approximations can provide satisfactory estimates for operational purposes.
The work presented here has been extended in two directions. Firstly, a more general sensitivity analysis of $\bar{U}$ has been performed. Secondly, we have assessed another key performance index for data center architects and managers, namely the average data loss per year. These results will be discussed elsewhere [20].

Acknowledgments

We wish to thank our Orange colleagues Ruby Krishnaswamy for stimulating discussions, and Christian Bourliataud and Thomas Rivera for access to their computing facility.

References

[1] Bhagwat, D., Pollack, K., Long, D.D.E., Schwarz, T., Miller, E. L. & Pâris, J.F. (2006). Providing High Reliability in a Minimum Redundancy Archival Storage System. *Proc. of the 14th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS 2006)*, 413-421.
[2] Dai, Y.S., Yang, B., Dongarra, J. & Zhang, G. (2009). Cloud Service Reliability: Modeling and Analysis. *Proc. of the 15th IEEE Pacific Rim International Symposium on Dependable Computing.*
[3] Dhillon, B.S. (2005). *Reliability, Quality, and Safety for Engineers*. CRC Press, 2000 N.W. Corporate Blvd., Boca Raton, Florida.
[4] Druault-Vicard, A. & Tanguy, C. (2008). Exact Failure Frequency Calculations for Extended Systems. *http://arxiv.org/abs/cs/0612141*. (unpublished)

[5] Elerath, J.G. & Pecht, M. (2007). Enhanced Reliability Modeling of RAID Storage Systems. *Proc. of the 37th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN'07) pp. 175-184*.

[6] Greenan, K., Plank, J.S. & Wylie, J.J. (2010). Mean time to meaningless: MTTDL, Markov models, and storage system reliability. *In Proc. of the Second Workshop on Hot Topics in Storage and File Systems.*

[7] Hafner, J.L. & Rao, K.K. (2006). Notes on Reliability Models for Non-MDS Erasure Codes. IBM Research Division, Technical Report RJ10391 (A0610-035), October 2006.

[8] Huang C., Simitci H., Xu Y., Ogus A., Calder B., Gopalan P., Li J & Yekhanin S. (2012). Erasure Coding in Windows Azure Storage. *Proc. of the 2012 USENIX Annual Technical Conference (ATC'12).*

[9] Iliadis, I. & Venkatesan, V. (2014). Expected Annual Fraction of Data Loss as a Metric for Data Storage Reliability. *Proc. of the 22nd International Symposium on Modelling, Analysis Simulation of Computer and Telecommunication Systems (MASCOTS 2014) pp. 375-384*.

[10] Jiang, W., Hu, C., Zhou, Y. & Kanevsky, A. (2008). Are Disks the Dominant Contributor for Storage Failures? A Comprehensive Study of Storage Subsystem Failure Characteristics. *ACM Transactions on Storage* 4, 3, Article 7.

[11] Kuo, W. & Zuo, M. J. (2003). *Optimal Reliability Modeling*. John Wiley & Sons, Inc., Hoboken, New Jersey.

[12] Leslie, M., Davies, J. & Huffman, T. (2006). A Comparison of Replication Strategies for Reliable Decentralised Storage. *Journal of Networks*, 1, 6, 36-44.

[13] Lian, Q., Chen, W. & Zhang, Z. (2005). On the Impact of Replica Placement to the Reliability of Distributed Brick Storage Systems. *Proc. of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS 2005),* 187-196.

[14] Pinheiro, E., Weber, W.-D. & Barroso, L. A. (2007). Failure Trends in a Large Disk Drive Population. *Proceedings of the USENIX Conference on File and Storage Technologies,* 17-28.

[15] Rao, K.K., Hafner, J.L. & Golding, R.A. (2011). Reliability for Networked Storage Nodes. *IEEE Transactions on Dependable and Secure Computing*, 8, 3, 404-418.

[16] Rausand, M. & Høyland, A. (2004). *System Reliability Theory: Models, Statistical Methods, and Applications*. John Wiley & Sons, Inc., Hoboken, New Jersey.

[17] Rosenthal, D.S.H. (2010). Bit Preservation: A Solved Problem? *International Journal of Digital Curation 5(1),* 134-148.

[18] Schroeder, B. & Gibson, G.A. (2006). Understanding disk failure rates: What does an MTTF of 1,000,000 hours mean to you? *ACM Transactions on Storage*, 3, 3, Article 8.

[19] Sun, D.W., Chang, G.R., Gao, S., Jin, L.Z. & Wang, X.W. (2012). Modeling a Dynamic Data Replication Strategy to Increase System Availability in Cloud Computing Environments. *Journal of Computer Science and Technology* 27, 2, 256-272.

[20] Tanguy, C., Besson, M., Krishnaswamy, R. & Grall, A. (2015). On data unavailability and file loss in coded data storage systems for the Cloud, *submitted to ESREL 2015*.

[21] Thomasian, A. & Blaum, M. (2012). Mirrored Disk Organization Reliability Analysis. *IEEE Transactions on Computers* 55, 12, 1640-1644.

[22] Venkatesan, V. & Iliadis, I. (2012). A General Reliability Model for Data Storage Systems. *Proc. of the Ninth International Conference on Quantitative Evaluation of Systems (QEST 2012),* 209-219.

[23] Venkatesan, V. & Iliadis, I. (2013). Effect of Codeword Placement on the Reliability of Erasure Coded Data Storage Systems. In *Quantitative Evaluation of Systems, Lecture Notes in Computer Science, vol. 8054, editors K. Joshi, M. Siegle, M. Stoelinga & P. R. D'Argenio, Springer:* 241-257.

[24] Venkatesan, V., Iliadis, I., Fragouli C. & Urbanke, R. (2011). Reliability of Clustered vs. Declustered Replica Placement in Data Storage Systems. *Proc. of the 19th Annual IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems,* 307-317.