Reliability of Erasure Coded Systems under Rebuild Bandwidth Constraints

Ilias Iliadis
IBM Research – Zurich
8803 Rüschlikon, Switzerland
Email: ili@zurich.ibm.com

***Abstract*—Modern storage systems employ erasure coding redundancy and recovering schemes to ensure high data reliability at high storage efficiency. The widely used replication scheme belongs to this broad class of erasure coding schemes. The effectiveness of these schemes has been evaluated based on the Mean Time to Data Loss** (MTTDL) **and the Expected Annual Fraction of Data Loss** (EAFDL) **metrics. To improve the reliability of data storage systems, certain data placement and rebuild schemes reduce the rebuild times by recovering data in parallel from the storage devices. It is often assumed though that there is sufficient network bandwidth to transfer the data required by the rebuild process at full speed. In large-scale data storage systems, however, the network bandwidth is constrained. This article obtains the** MTTDL **and** EAFDL **of erasure coded systems analytically for the symmetric, clustered, and declustered data placement schemes under network rebuild bandwidth constraints. The resulting reliability degradation is assessed and the results obtained establish that the declustered placement scheme offers superior reliability in terms of both metrics. Efficient codeword configurations that achieve high reliability in the presence of network rebuild bandwidth constraints are identified.**

***Keywords*–Storage; Reliability; Data placement; MTTDL; EAFDL; RAID; MDS codes; Information Dispersal Algorithm; Prioritized rebuild; Repair bandwidth; Network bandwidth constraint.**

I. Introduction

In today's large-scale data storage systems, data redundancy is introduced to ensure that data lost owing to device and component failures can be recovered. Appropriate redundancy schemes are deployed to prevent permanent loss of data and, consequently, enhance the reliability of storage systems. The effectiveness of these schemes has been evaluated based on the Mean Time to Data Loss (MTTDL) [1-20] and, more recently, the Fraction of Data Loss Per Year (FDLPY) [21] and the equivalent Expected Annual Fraction of Data Loss (EAFDL) reliability metrics [22-24]. Analytical reliability expressions for the MTTDL were obtained predominately using Markovian models, which assume that component failure and rebuild times are independent and exponentially distributed. In practice though, these distributions are not exponential. To cope with this issue, system reliability was assessed in [16][18][23][24] using an alternative methodology that does not involve any Markovian analysis and considers the practical case of non-exponential failure and rebuild time distributions. Moreover, the misconception reported in [25] that MTTDL derivations based on Markovian models provide unrealistic results was dispelled in [26] by invoking improved MTTDL derivations that yield satisfactory results, and also by drawing on prior work that analytically obtains MTTDL without involving any Markovian analysis.

Earlier works have predominately considered the MTTDL metric, whereas recent works have also considered the EAFDL metric [22][23][24]. The introduction of the latter metric was motivated by the fact that Amazon S3 considers the durability of data over a given year [27], and, similarly, Facebook [28], LinkedIn [29] and Yahoo! [30] consider the amount of data lost in given periods.

To protect data from being lost and improve the reliability of data storage systems, replication-based storage systems spread replicas corresponding to data stored on each storage device across several other storage devices. To improve the low storage efficiency associated with the replication schemes, erasure coding schemes that provide a high data reliability as well as a high storage efficiency are deployed. Special cases of such codes are the Redundant Arrays of Inexpensive Disks (RAID) schemes, such as RAID-5 and RAID-6, that have been extensively deployed in the past thirty years [1][2].

State-of-the-art data storage systems [31-34] employ more general erasure codes that affect the reliability, performance, and the storage and reconstruction overhead of the system. In this article, we focus on the reliability assessment of erasure coded systems in terms of the MTTDL and EAFDL metrics. These metrics were analytically derived in [23] for the symmetric, clustered, and declustered data placement schemes under the assumption that there is sufficient network bandwidth to transfer the data required by the rebuild process at full speed. For instance, in the case of a declustered placement, redundant data associated with the data stored on a given device is placed across all remaining devices in the system. In this way, the rebuild process can be parallelized, which in turn results in short rebuild times. The restoration time can be minimized provided there is sufficient network rebuild bandwidth available. In large-scale data storage systems though, the network bandwidth is constrained.

The effect of network rebuild bandwidth constraints on the reliability of replication-based storage systems was studied in [8][15]. It was found that spreading replicas over a higher number of devices than what the network rebuild bandwidth can support at full speed during a parallel rebuild process, led to system reliability being significantly reduced. The reliability of erasure coded systems in the absence of bandwidth constraints was assessed in [23]. The MTTDL and EAFDL metrics were obtained analytically for the symmetric, clustered, and declustered data placement schemes based on a general framework and methodology. In this article, we recognize that this methodology also holds in the case of network rebuild bandwidth constraints and apply it to derive enhanced closed-form reliability expressions for the MTTDL and EAFDL metrics for these placement schemes in the presence of such rebuild bandwidth constraints. Subsequently, we provide insight into the effect of the placement schemes and the impact of the available network rebuild bandwidth on system reliability. The validity of this methodology for accurately assessing the reliability of storage systems was confirmed by means of simulation in several contexts [14-16, 18, 22]. It was demonstrated that the theoretical predictions for the reliability of systems comprised of highly reliable storage devices match well with the simulation results obtained. Consequently, the emphasis of the present work is on the theoretical assessment of the effect of network rebuild bandwidth constraints on the reliability of erasure coded systems. Also, this work extends the reliability results obtained in [15] for the special case of replication-based storage systems to the more general case of erasure coded systems.

The remainder of the article is organized as follows. Section II describes the storage system model and the corresponding parameters considered. Section III presents the adaptation of a general framework and methodology for deriving the MTTDL and EAFDL metrics analytically for the case of erasure coded systems under network rebuild bandwidth constraints. Closed-form expressions for the symmetric, clustered, and declustered placement schemes are derived. Section IV presents numerical results demonstrating the effectiveness of the erasure coding redundancy schemes for improving the system reliability. It also assesses the sensitivity to the network rebuild bandwidth constraints under various codeword configurations. Section V provides a discussion on the applicability of the results obtained. Finally, we conclude in Section VI.

II. Storage System Model

Modern data storage systems use erasure coded schemes to protect data from device failures. When devices fail, the redundancy of the data affected is reduced and eventually lost. To avoid irrecoverable data loss, the system performs rebuild operations that use the data stored in the surviving devices to reconstruct the temporarily lost data, thus maintaining the initial data redundancy. We proceed by briefly reviewing the basic concepts of erasure coding and data recovery procedures of such storage systems. To assess their reliability, we consider the model used in [23], and adopt and extend the notation. More precisely, the storage system considered comprises n storage devices (nodes or disks), with each device storing an amount c of data, such that the total storage capacity of the system is $n\,c$.

A. Redundancy

User data is divided into blocks (or symbols) of a fixed size (e.g., sector size of 512 bytes) and complemented with parity symbols to form codewords. We consider (m, l) maximum distance separable (MDS) erasure codes, which are a mapping from l user data symbols to a set of m $(> l)$ symbols, called a codeword, having the property that any subset containing l of the m symbols of the codeword can be used to decode (reconstruct, recover) the codeword. The corresponding storage efficiency, s_{eff}, is given by

$$s_{\text{eff}} = \frac{l}{m} \,. \tag{1}$$

TABLE I. Notation of system parameters

Parameter	Definition
n	number of storage devices
c	amount of data stored on each device
l	number of user-data symbols per codeword $(l \geq 1)$
m	total number of symbols per codeword $(m > l)$
(m, l)	MDS-code structure
k	spread factor of the data placement scheme, or group size (number of devices in a group)
b	reserved rebuild bandwidth per device
B_{max}	maximum network rebuild bandwidth
$F_\lambda(.)$	cumulative distribution function of device lifetimes
s_{eff}	storage efficiency of redundancy scheme $(s_{\text{eff}} = l/m)$
U	amount of user data stored in the system $(U = s_{\text{eff}}\, n\, c)$
$\tilde{r}$	minimum number of codeword symbols lost that lead to an irrecoverable data loss $(\tilde{r} = m - l + 1$ and $2 \leq \tilde{r} \leq m)$
N_b	maximum number of devices from which rebuild can occur at full speed in parallel $(N_b = B_{\text{max}}/b)$
B_{eff}	effective network rebuild bandwidth
$1/\mu$	time to read (or write) an amount c of data at a rate b from (or to) a device $(1/\mu = c/b)$
$1/\lambda$	mean time to failure of a storage device $(1/\lambda = \int_0^\infty [1 - F_\lambda(t)] dt)$

Consequently, the amount of user data, U, stored in the system is given by

$$U = s_{\text{eff}}\; n\, c \;=\; \frac{l\, n\, c}{m} \,. \tag{2}$$

The notation used is summarized in Table I. The parameters are divided according to whether they are independent or derived, and are listed in the upper and the lower part of the table, respectively.

The m symbols of each codeword are stored on m distinct devices, such that the system can tolerate any $\tilde{r} - 1$ device failures, but $\tilde{r}$ device failures may lead to data loss, with

$$\tilde{r} = m - l + 1 \,. \tag{3}$$

From the preceding, it follows that

$$1 \leq l < m \qquad \text{and} \qquad 2 \leq \tilde{r} \leq m \,. \tag{4}$$

Examples of MDS erasure codes are the following:

Replication: A replication-based system with a replication factor r can tolerate any loss of up to $r - 1$ copies of some data, such that $l = 1$, $m = r$ and $\tilde{r} = r$. Also, its storage efficiency is equal to $s_{\text{eff}}^{(\text{replication})} = 1/r$.
RAID-5: A RAID-5 array comprised of N devices uses an $(N, N-1)$ MDS code, such that $l = N - 1$, $m = N$ and $\tilde{r} = 2$. It can therefore tolerate the loss of up to one device, and its storage efficiency is equal to $s_{\text{eff}}^{(\text{RAID-5})} = (N-1)/N$.
RAID-6: A RAID-6 array comprised of N devices uses an $(N, N-2)$ MDS code, such that $l = N - 2$, $m = N$ and $\tilde{r} = 3$. It can therefore tolerate a loss of up to two devices, and its storage efficiency is equal to $s_{\text{eff}}^{(\text{RAID-6})} = (N-2)/N$.
Reed–Solomon: It is based on (m, l) MDS erasure codes.

B. Symmetric Codeword Placement

According to a symmetric codeword placement, each codeword is stored on m distinct devices with one symbol per device. In a large storage system, the number of devices, n, is usually much larger than the codeword length, m. Therefore, there are many ways in which a codeword of m symbols can be stored across a subset of the n devices. For each device in the system, the *redundancy spread factor* k denotes the number of devices over which the codewords stored on that device are spread [18]. The system effectively comprises n/k

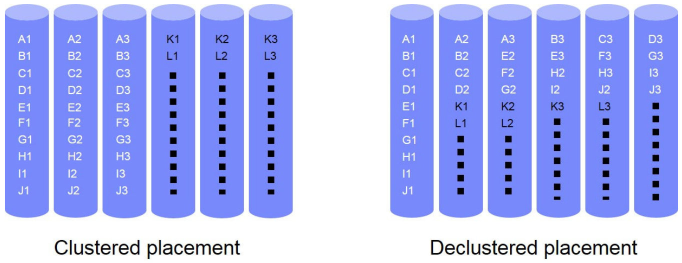


Figure 1. Clustered and declustered placement of codewords of length $m = 3$ on $n = 6$ devices. X1, X2, X3 represent a codeword (X = A, B, C, . . . , L).

disjoint groups of k devices. Each group contains an amount U/k of user data, with the corresponding codewords placed on the corresponding k devices in a distributed manner. Each codeword is placed entirely in one of the n/k groups. Within each group, all $\binom{k}{m}$ possible ways of placing m symbols across k devices are equally used to store all the codewords in that group.

In such a symmetric placement scheme, within each of the n/k groups, the $m-1$ codeword symbols corresponding to the data on each device are *equally* spread across the remaining $k-1$ devices, the $m-2$ codeword symbols corresponding to the codewords shared by any two devices are equally spread across the remaining $k-2$ devices, and so on. Note also that the n/k groups are logical and therefore need not be physically located in the same node/rack/datacenter.

We proceed by considering the clustered and declustered placement schemes, which are special cases of symmetric placement schemes for which k is equal to m and n, respectively. This results in n/m groups for clustered and one group for declustered placement schemes.

1) Clustered Placement: The n devices are divided into disjoint sets of m devices, referred to as *clusters*. According to the *clustered* placement, each codeword is stored across the devices of a particular cluster, as shown in Figure 1. In such a placement scheme, it can be seen that no cluster stores the redundancies that correspond to data stored on another cluster. The entire storage system can essentially be modeled as consisting of n/m independent clusters. In each cluster, data loss occurs when $\tilde{r}$ devices fail successively before rebuild operations complete successfully.

2) Declustered Placement: In this placement scheme, all $\binom{n}{m}$ possible ways of placing m symbols across n devices are equally used to store all the codewords in the system, as shown in Figure 1.

The clustered and declustered placement schemes represent the two extremes in which the symbols of the codewords associated with the data stored on a failing device are spread across the remaining devices and hence the extremes of the degree of parallelism that can be exploited when rebuilding this data. For declustered placement, the symbols are spread equally across *all* remaining devices, whereas for clustered placement, the symbols are spread across the smallest possible number of devices.

C. Codeword Reconstruction

When storage devices fail, codewords lose some of their symbols, and this leads to a reduction in data redundancy. The system attempts to maintain its redundancy by reconstructing the lost codeword symbols using the surviving symbols of the affected codewords.

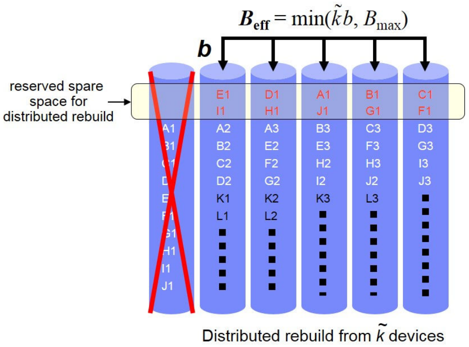

$B_{\text{eff}} = \min(\tilde{k}b, B_{\max})$
b
reserved spare space for distributed rebuild
Distributed rebuild from $\tilde{k}$ devices

Figure 2. Rebuild under declustered placement.

When a declustered placement scheme is used, as shown in Figure 2, spare space is reserved on each device for temporarily storing the reconstructed codeword symbols before they are transferred to a new replacement device. The rebuild process used to restore the data lost by failed devices is assumed to be both *prioritized* and *distributed*. As discussed in [23], a prioritized (or intelligent) rebuild process always attempts to first rebuild the *most-exposed* codewords, namely, the codewords that have lost the largest number of symbols. The prioritized rebuild process recovers one of the symbols that each of the most-exposed codewords has lost by reading $m - \tilde{r} + 1$ of the remaining symbols. In a distributed rebuild process, the codeword symbols lost by failed devices are reconstructed by reading surviving symbols from a number, say $\tilde{k}$, of surviving devices, and storing the recovered symbols in the reserved spare space of the $\tilde{k}$ surviving devices, as shown in Figure 2.

A certain proportion of the device bandwidth is reserved for data recovery during the rebuild process, with b denoting the actual reserved rebuild bandwidth per device. This bandwidth is usually only a fraction of the total bandwidth available at each device, with the remaining bandwidth being used to serve user requests. Thus, the lost symbols are rebuilt in parallel using the rebuild bandwidth b available on each surviving device. During this process, it is desirable to reconstruct the lost codeword symbols on devices in which another symbol of the same codeword is not already present. Assuming that the system is at exposure level u (as described in Section II-D below) b_u $(\leq b)$ denotes the rate at which the amount of data that needs to be rebuilt (repair traffic) is written to selected device(s). In particular, $1/\mu$ denotes the time required to read (or write) an amount c of data from (or to) a device, given by

$$\frac{1}{\mu} = \frac{c}{b} \,. \tag{5}$$

In a distributed rebuild process involving $\tilde{k}$ devices, the total network bandwidth required to perform rebuild at full speed is $\tilde{k}\,b$. Let $B_{\max}$ $(\geq b)$ denote the maximum available network bandwidth for rebuilds. Then, the effective network rebuild bandwidth used by rebuilds, $B_{\text{eff}}(\tilde{k})$, cannot exceed $B_{\max}$ and is therefore given by

$$B_{\text{eff}}(\tilde{k}) = \min(\tilde{k}\,b, B_{\max}) = \min(\tilde{k}, N_b)\,b \,, \tag{6}$$

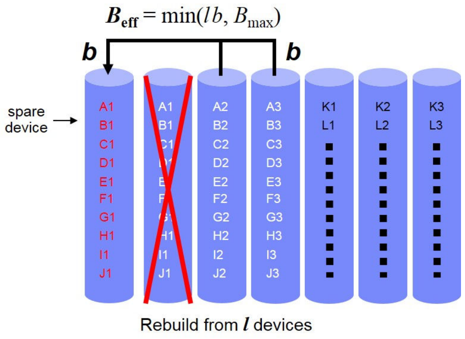

$B_{\text{eff}} = \min(lb, B_{\max})$
b
b
spare device
A1 B1 C1 D1 E1 F1 G1 H1 I1 J1
A1 B1 C1 D1 E1 F1 G1 H1 I1 J1
A2 B2 C2 D2 E2 F2 G2 H2 I2 J2
A3 B3 C3 D3 E3 F3 G3 H3 I3 J3
K1 L1
K2 L2
K3 L3
Rebuild from l devices

Figure 3. Rebuild under clustered placement.

where N_b specifies the effective maximum number of devices from which rebuild can occur in parallel at full speed, and is given by

$$N_b \triangleq \frac{B_{\max}}{b} \, . \tag{7}$$

Note that N_b may not be an integer; it only represents the *effective* maximum number of devices from which distributed rebuild can occur at full speed. Substituting $b = c\mu$ into (6), we get

$$B_{\text{eff}} = \min(\tilde{k}, N_b)\, c\mu \, . \tag{8}$$

A similar reconstruction process is used for other symmetric placement schemes within each group of k devices, except for the clustered placement. When clustered placement is used, the codeword symbols are spread across all $k = m$ devices in each group (cluster). Therefore, reconstructing the lost symbols on the surviving devices of a group will result in more than one symbol of the same codeword on the same device. To avoid this, the lost symbols are reconstructed directly in spare devices as shown in Figure 3. In these reconstruction processes, decoding and re-encoding of data are assumed to be done on the fly and so the time taken for reconstruction is equal to the time taken to read and write the required data to the devices. Note also that alternative erasure coding schemes have been proposed to reduce the amount of data transferred over the storage network during reconstruction (see [35][36] and references therein).

D. Exposure Levels and Amount of Data to Rebuild

At time t, $D_j(t)$ denotes the number of codewords that have lost j symbols, with $0 \le j \le \tilde{r}$. The system is at exposure level u $(0 \le u \le \tilde{r})$, where

$$u = \max_{D_j(t)>0} j. \tag{9}$$

The system is at exposure level u if there are codewords with $m - u$ symbols left, but there are no codewords with fewer than $m - u$ symbols left in the system, that is, $D_u(t) > 0$, and $D_j(t) = 0$, for all $j > u$. These codewords are referred to as the *most-exposed* codewords. At $t = 0$, $D_j(0) = 0$, for all $j > 0$, and $D_0(0)$ is the total number of codewords stored in the system. Device failures and rebuild processes cause the values of $D_1(t), \cdots, D_{\tilde{r}}(t)$ to change over time, and when a data loss occurs, $D_{\tilde{r}}(t) > 0$. Device failures cause transitions to higher exposure levels, whereas rebuilds cause transitions to lower ones. Let t_u denote the time of the first transition from exposure level $u - 1$ to exposure level u, and t_u^+ the instant immediately after t_u. Then, the number, C_u, of most exposed codewords when entering exposure level u, $u = 1, \ldots, \tilde{r}$, is given by $C_u = D_u(t_u^+)$.

Analytic expressions for the reliability metrics of interest were derived in [23], using the direct path approximation, which considers only transitions from lower to higher exposure levels [14][16][18]. This implies that each exposure level is entered only once.

E. Failure and Rebuild Time Distributions

We adopt the model and notation considered in [24]. The lifetimes of the n devices are assumed to be independent and identically distributed, with a cumulative distribution function $F_\lambda(.)$ and a mean of $1/\lambda$. Real-world distributions, such as Weibull and gamma, as well as exponential distributions that belong to the large class defined in [16] are considered. The storage devices are characterized to be *highly reliable* in that the ratio of the mean time $1/\mu$ to read all contents of a device (which typically is on the order of tens of hours), to the mean time to failure of a device $1/\lambda$ (which is typically at least on the order of thousands of hours) is small, that is,

$$\frac{\lambda}{\mu} = \frac{\lambda\, c}{b} \ll 1 \, . \tag{10}$$

We consider storage devices whose the cumulative distribution function F_λ satisfies the condition

$$\mu \int_0^{1/\mu} F_\lambda(t)\, dt \ll 1, \quad \text{with } \frac{\lambda}{\mu} \ll 1 \, , \tag{11}$$

such that the MTTDL and EAFDL reliability metrics of erasure coded storage systems tend to be insensitive to the device failure distribution, that is, they depend only on its mean $1/\lambda$, but not on its density $F_\lambda(.)$[23].

III. Derivation of MTTDL and EAFDL

The MTTDL metric assesses the expected amount of time until some data can no longer be recovered and therefore is irrecoverably lost, whereas the EAFDL metric assesses the fraction of stored data that is expected to be lost by the system annually. The MTTDL$(B_{\max})$ and EAFDL$(B_{\max})$ metrics are derived as a function of $B_{\max}$ based on the framework and methodology presented in [23]. More specifically, this methodology uses the direct path approximation and does not involve any Markovian analysis. It holds for general failure time distributions, which can be exponential or non-exponential, such as the Weibull and gamma distributions that satisfy condition (11). Note that this framework is general in that it also applies in the case where the network rebuild bandwidth is constrained. The only parameters that are affected by the network rebuild bandwidth constraint are the rebuild rates and, accordingly, those parameters that depend on them, such as the rebuild times. Analytic expressions for the two metrics of interest were derived in [23, Equations (44) and (45)] as follows:

$$\text{MTTDL}(B_{\max}) \approx \frac{1}{n\,\lambda}\, \frac{(\tilde{r}-1)!}{(\lambda\, c)^{\tilde{r}-1}} \prod_{u=1}^{\tilde{r}-1} \frac{b_u(B_{\max})}{\tilde{n}_u}\, \frac{1}{V_u^{\tilde{r}-1-u}} \, , \tag{12}$$

and

$$\text{EAFDL}(B_{\text{max}}) \approx m\,\lambda\,(\lambda\,c)^{\tilde{r}-1}\,\frac{1}{\tilde{r}\,!}\prod_{u=1}^{\tilde{r}-1}\frac{\tilde{n}_u}{b_u(B_{\text{max}})}\,V_u^{\tilde{r}-u}\ , \tag{13}$$

where $\tilde{n}_u$ represents the number of devices at exposure level u whose failure before the rebuild of the most-exposed codewords causes an exposure level transition to level $u+1$. Also, V_u represents the fraction of the most-exposed codewords at exposure level u that have symbols stored on a newly failed device that causes the exposure level transition $u \to u{+}1$. Note that this fraction depends only on the codeword placement scheme. As mentioned in the preceding, b_u, the rate at which the amount of data that needs to be rebuilt at exposure level u is written to selected device(s), depends on B_{max}, the maximum network rebuild bandwidth.

Remark 1: From [23, Equation (43)], it follows that the expected amount $E(H)$ of data lost, given that a data loss has occurred, does not depend on b_u and therefore is not affected by the maximum network rebuild bandwidth. Consequently, this reliability metric is not considered in this article.

Remark 2: The analytic expressions for the MTTDL and EAFDL reliability metrics were derived in [23] in the absence of network rebuild bandwidth constraints. Consequently, they correspond to the case of $B_{\text{max}} = \infty$, with the two metrics being denoted by $\text{MTTDL}(\infty)$ and $\text{EAFDL}(\infty)$, respectively.

From (12) and (13), it follows that

$$\frac{\text{MTTDL}(B_{\text{max}})}{\text{MTTDL}(\infty)} = \frac{\text{EAFDL}(\infty)}{\text{EAFDL}(B_{\text{max}})} = \theta\ , \tag{14}$$

where θ represents the *reliability reduction factor* that assesses the reliability degradation due to a network rebuild bandwidth constraint, and is given by

$$\theta \triangleq \prod_{u=1}^{\tilde{r}-1}\frac{b_u(B_{\text{max}})}{b_u(\infty)}\ . \tag{15}$$

Remark 3: From (15), and given that $b_u(B_{\text{max}})$ decreases as B_{max} decreases, it follows that θ decreases as $\tilde{r}$ increases and B_{max} decreases.

A. Symmetric Placement

We consider the case where the redundancy spread factor k is in the interval $m < k \le n$. As discussed in [23, Section III-B], at each exposure level u, the *prioritized* rebuild process recovers one of the u symbols that each of the most-exposed codewords has lost by reading $m - \tilde{r} + 1$ of the remaining symbols from the $\tilde{n}_u$ surviving devices in the affected group. According to [23, Equation (46)], it holds that

$$\tilde{n}_u^{\text{sym}} = k - u\ . \tag{16}$$

Furthermore, in the absence of a network rebuild bandwidth constraint, the total write bandwidth, which is also the rebuild rate b_u, is given by [23, Equation (47)]

$$b_u^{\text{sym}}(\infty) = \frac{\tilde{n}_u^{\text{sym}}}{m-\tilde{r}+2}\,b \overset{(16)}{=} \frac{(k-u)\,b}{m-\tilde{r}+2}\ , \quad u = 1,\ldots,\tilde{r}-1\ . \tag{17}$$

In the presence though of a network rebuild bandwidth constraint, B_{max}, and according to (6), with $\tilde{k} = \tilde{n}_u = \tilde{n}_u^{\text{sym}}$, the rebuild rate b_u is given as a function of B_{max} by

$$b_u^{\text{sym}}(B_{\text{max}}) = \frac{B_{\text{eff}}(\tilde{n}_u)}{m-\tilde{r}+2} = \frac{\min(\tilde{n}_u\,b, B_{\text{max}})}{m-\tilde{r}+2} = \frac{\min(\tilde{n}_u, N_b)\,b}{m-\tilde{r}+2} \overset{(16)}{=} \frac{\min(k-u, N_b)\,b}{m-\tilde{r}+2}\ ,\ \text{ for }\ u = 1,\ldots,\tilde{r}-1\ . \tag{18}$$

Substituting (17) and (18) into (15) yields

$$\theta^{\,\text{sym}} = \prod_{u=1}^{\tilde{r}-1}\frac{\min(k-u, N_b)}{k-u}\ . \tag{19}$$

Note that when $N_b \ge k-1$, the system reliability is not affected because all rebuilds are performed at full speed, and therefore the θ factor is equal to one. However, when $N_b < k-1$, it may not be possible for some of the rebuilds to be performed at full speed, and therefore the factor θ will be less than one, which affects the system reliability. Consequently, the reliability reduction factor, θ, depends on the *bandwidth constraint factor*, ϕ, given by

$$\phi \triangleq \min\left(\frac{N_b}{k}, 1\right) \overset{(7)}{=} \min\left(\frac{B_{\text{max}}}{k\,b}, 1\right)\ ,\ \text{ with } 0 \le \phi \le 1\ . \tag{20}$$

From (19) and (20), and recognizing that $\min(k-u, N_b) = \min(\min(k-u,k), N_b) = \min(k-u, \min(k, N_b)) = \min(k\min(1, N_b/k), k-u) = \min(k\,\phi, k-u)$, it follows that

$$\theta^{\,\text{sym}} = \prod_{u=1}^{\tilde{r}-1}\min\left(\frac{\phi}{1-\frac{u}{k}}, 1\right)\ . \tag{21}$$

Using (3) and (21), and the fact that $\text{MTTDL}(\infty)$ and $\text{EAFDL}(\infty)$ are given by [23, Equations (49) and (50)], respectively, (14) yields

$$\text{MTTDL}_k^{\text{sym}}(B_{\text{max}}) \approx \frac{1}{n\,\lambda}\left[\frac{b}{(l+1)\,\lambda\,c}\right]^{m-l}(m-l)!\ \prod_{u=1}^{m-l}\left(\frac{k-u}{m-u}\right)^{m-l-u}\prod_{u=1}^{m-l}\min\left(\frac{\phi}{1-\frac{u}{k}}, 1\right)\ , \tag{22}$$

and

$$\text{EAFDL}_k^{\text{sym}}(B_{\text{max}}) \approx \lambda\left[\frac{(l+1)\,\lambda\,c}{b}\right]^{m-l}\frac{m}{(m-l+1)!}\ \prod_{u=1}^{m-l}\left(\frac{m-u}{k-u}\right)^{m-l+1-u}\Bigg/\prod_{u=1}^{m-l}\min\left(\frac{\phi}{1-\frac{u}{k}}, 1\right)\ , \tag{23}$$

where B_{max} is expressed via ϕ given by (20).

Note that for a replication-based system, for which $m = r$ and $l = 1$, and by virtue of (19) and (21), (22) is in agreement with Equation (24) of [15], with $c/b = 1/\mu$.

Remark 4: From (22) and (23), it follows that $\text{MTTDL}_k^{\text{sym}}$ depends on n, but $\text{EAFDL}_k^{\text{sym}}$ does not.

Remark 5: From (22) and (23), and for any value of ϕ, it can be proved that for $m{-}l \ge 2$, $\text{MTTDL}_k^{\text{sym}}$ is increasing in k. It can also be proved that for any $m-l \ge 1$, $\text{EAFDL}_k^{\text{sym}}$ is not increasing in k. Consequently, within the class of symmetric placement schemes considered, that is, for $l{+}1 < m < k \le n$, the $\text{MTTDL}_k^{\text{sym}}$ is maximized and the $\text{EAFDL}_k^{\text{sym}}$ is minimized by the declustered placement scheme, that is, when $k = n$.

B. Clustered Placement

In the clustered placement scheme, the n devices are divided into disjoint sets of m devices, referred to as *clusters*. According to the *clustered* placement, each codeword is stored across the devices of a particular cluster. At each exposure level u, the rebuild process recovers one of the u symbols that each of the C_u most-exposed codewords has lost by reading $m - \tilde{r} + 1$ of the remaining symbols. Note that the remaining symbols are stored on the $m - u$ surviving devices in the affected group. According to [23, Equation (53)], it holds that

$$\tilde{n}_u^{\text{clus}} = m - u \; . \tag{24}$$

In the case of clustered placement, the rebuild process recovers the lost symbols by reading l symbols from l of the $\tilde{n}_u$ surviving devices of the affected cluster. In the absence of a network rebuild bandwidth constraint, the symbols are read at a rate of b from each of the l devices, such that the effective network rebuild bandwidth is equal to $B_{\text{eff}} = l\, b$. Subsequently, the lost symbols are computed on-the-fly and written to a spare device at a rate of $B_{\text{eff}}/l = b$. Consequently, it holds that

$$b_u^{\text{clus}}(\infty) = b \, , \quad u = 1, \ldots, \tilde{r} - 1 \; . \tag{25}$$

In the presence though of a network rebuild bandwidth constraint, $B_{\max}$, the effective network rebuild bandwidth is equal to $B_{\text{eff}} = \min(l\, b, B_{\max})$, which implies that the lost symbols are written to a spare device at a rate of B_{eff}/l. Thus, the rebuild rate b_u is given as a function of $B_{\max}$ by

$$b_u^{\text{clus}}(B_{\max}) = \frac{B_{\text{eff}}(B_{\max})}{l} = \frac{\min(l\, b, B_{\max})}{l} = \frac{\min(l, N_b)\, b}{l} \, , \quad \text{for } \; u = 1, \ldots, \tilde{r} - 1 \; . \tag{26}$$

Substituting (25) and (26) into (15) yields

$$\theta^{\,\text{clus}} \; = \; \left(\frac{\min(l, N_b)}{l} \right)^{\tilde{r}-1} \; . \tag{27}$$

As $l < m$, it holds that $\min(l, N_b) = \min(\min(l, m), N_b) = \min(\min(N_b, m), l) = \min(m \min(N_b/m, 1), l) = \min(m\phi, l)$, where, analogously to (20), and with $k = m$,

$$\phi \triangleq \min\left(\frac{N_b}{m}, 1 \right) \overset{(7)}{=} \min\left(\frac{B_{\max}}{m\, b}, 1 \right) \, , \quad \text{where } 0 \le \phi \le 1 \; . \tag{28}$$

Consequently, (27) yields

$$\theta^{\,\text{clus}} \; = \; \min\left(\frac{m}{l}\, \phi \, , 1 \right)^{\tilde{r}-1} \; . \tag{29}$$

Remark 6: From (29), it follows that for $m\, \phi/l \ge 1$ or, equivalently, for $\phi \ge s_{\text{eff}} = l/m$, $\theta^{\,\text{clus}}$ is equal to one, which implies that the bandwidth constraint does not affect the system reliability.

Using (3) and (29), and the fact that $\text{MTTDL}(\infty)$ and $\text{EAFDL}(\infty)$ are given by [23, Equations (56) and (57)], respectively, (14) yields

$$\text{MTTDL}^{\text{clus}}(B_{\max}) \approx \; \frac{1}{n\, \lambda} \left(\frac{\min(m\, \phi, \, l)\, b}{l\, \lambda\, c} \right)^{m-l} \frac{1}{\binom{m-1}{l-1}} \, , \tag{30}$$

$$\text{EAFDL}^{\text{clus}}(B_{\max}) \approx \; \lambda \left(\frac{l\, \lambda\, c}{\min(m\, \phi, \, l)\, b} \right)^{m-l} \binom{m}{l-1} \, , \tag{31}$$

where $B_{\max}$ is expressed via ϕ given by (28).

Remark 7: Note that as far as the data placement is concerned, the clustered placement scheme is a special case of a symmetric placement scheme for which k is equal to m. However, its reliability assessment cannot be directly obtained from the reliability results derived in Section III-A for the symmetric placement scheme by simply setting $k = m$. The reason for that is the difference in the rebuild processes. In the case of a symmetric placement scheme, recovered symbols are written to the spare space of existing devices, whereas in the case of a clustered placement scheme, recovered symbols are written to a spare device. This results in different rebuild bandwidths, which are given by (17) and (25), respectively.

C. Declustered Placement

The declustered placement scheme is a special case of a symmetric placement scheme in which k is equal to n. Consequently, for $k = n$, (22) and (23) yield

$$\text{MTTDL}^{\text{declus}}(B_{\max}) \approx \; \frac{1}{n\, \lambda} \left[\frac{b}{(l+1)\, \lambda\, c} \right]^{m-l} (m-l)! \prod_{u=1}^{m-l} \left(\frac{n-u}{m-u} \right)^{m-l-u} \prod_{u=1}^{m-l} \min\left(\frac{\phi}{1 - \frac{u}{n}}, 1 \right) \, , \tag{32}$$

and

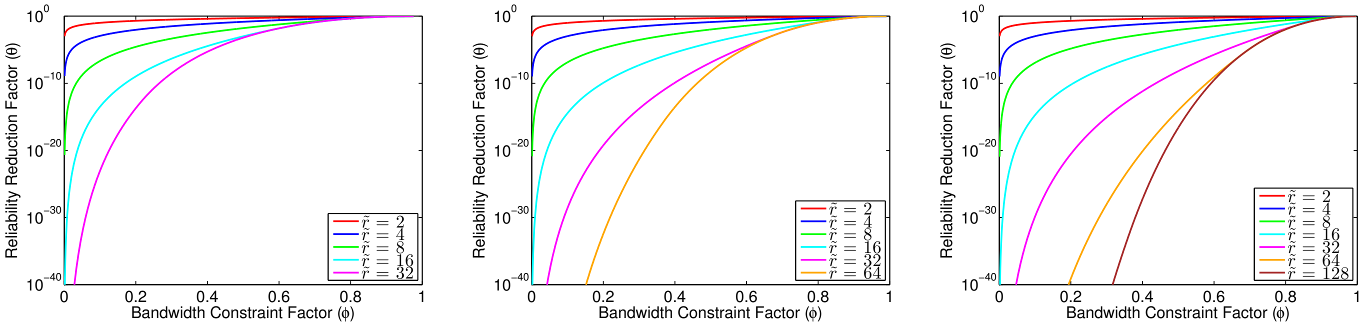
$$\text{EAFDL}^{\text{declus}}(B_{\max}) \approx \; \lambda \left[\frac{(l+1)\, \lambda\, c}{b} \right]^{m-l} \frac{m}{(m-l+1)!} \prod_{u=1}^{m-l} \left(\frac{m-u}{n-u} \right)^{m-l+1-u} \Bigg/ \prod_{u=1}^{m-l} \min\left(\frac{\phi}{1 - \frac{u}{n}}, 1 \right) \, , \tag{33}$$

where $B_{\max}$ is expressed via ϕ given by (20) with $k = n$.

IV. Numerical Results

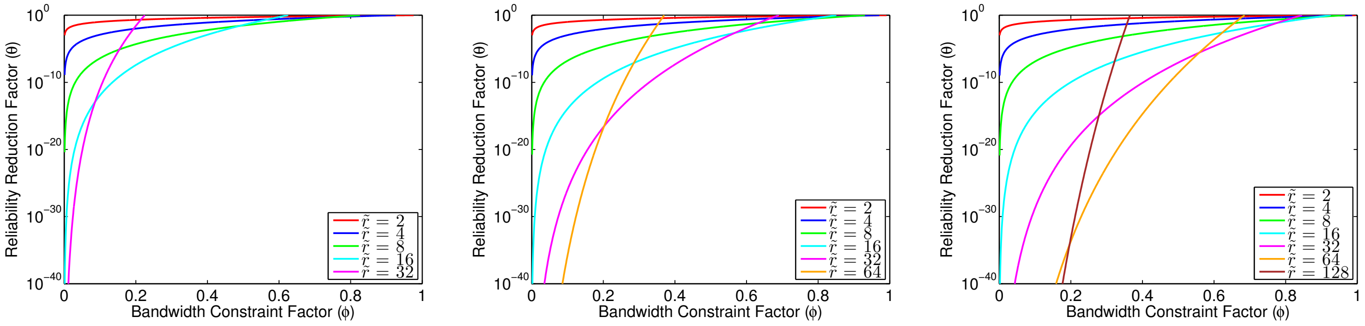
First, we assess the reduction in reliability owing to bandwidth constraints. The reliability reduction factor, θ, is obtained by (21) and (29) for the symmetric and clustered placements, respectively, and shown in Figures 4 and 5 as a function of the bandwidth constraint factor. For a symmetric placement scheme, Figure 4 demonstrates that as the group size k increases, the reliability reduction factor θ decreases and the magnitude of the reduction is more pronounced for larger values of $\tilde{r}$. Clearly, if codewords are spread over a higher number of devices than what the network rebuild bandwidth can support at full speed during a parallel rebuild process, the system reliability is affected and a drastic reliability degradation occurs as the system size increases. In contrast, according to Remark 6, the reliability of a clustered placement scheme remains unaffected for $\phi \ge l/m = (m - \tilde{r} + 1)/m$. This is due to the fact that the effective rebuild bandwidth is significantly smaller because the rebuilds are not distributed, but performed directly on a spare device. However, as Figure 5 demonstrates, for $\phi < l/m$, the reliability reduction factor drops sharply, especially for large values of $\tilde{r}$.

Next, we consider a storage system of a given size and assess its reliability for various codeword configurations, storage efficiencies, and network rebuild bandwidth constraints.



(a) $k = 40$ (b) $k = 100$ (c) $k = 200$

Figure 4. Reliability reduction factor vs. bandwidth constraint factor for various values of $\tilde{r}$; symmetric placement.



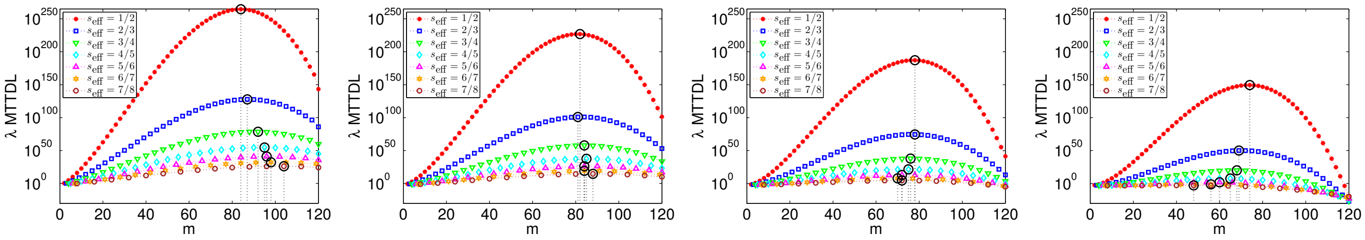
(a) $m = 40$ (b) $m = 100$ (c) $m = 200$

Figure 5. Reliability reduction factor vs. bandwidth constraint factor for various values of $\tilde{r}$; clustered placement.

In particular, we consider a system containing 120 devices under a declustered placement scheme $(k = n = 120)$, which according to Remark 5 is the optimal one within the class of symmetric schemes. The amount of user data stored, U, is determined by the storage efficiency, s_{eff}, via (2). As discussed in Section II-E, the analytical reliability results obtained are accurate when the storage devices are highly reliable, that is, when the ratio λ/μ of the mean rebuild time $1/\mu$ to the mean time to failure of a device $1/\lambda$ is very small. We proceed by considering systems for which it holds that $\lambda/\mu = \lambda\, c/b = 0.001$.

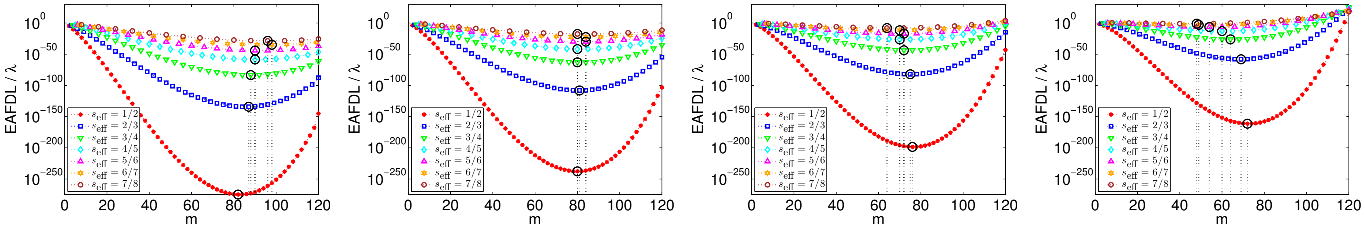
The combined effect of the network rebuild bandwidth constraint and the system efficiency on the normalized λ MTTDL measure is obtained by (32) and shown in Figure 6 as a function of the codeword length. In particular, when the codeword length is equal to the system size $(m = k = n)$, the placement becomes clustered and the normalized λ MTTDL measure is obtained by (30). Three cases for the network rebuild bandwidth constraint were considered: $\phi = 1$ corresponds to the case where there is no network rebuild bandwidth constraint given that $N_b \geq k = 120$ or, equivalently, $B_{\max} \geq k\, b = 120\, b$; $\phi = 0.1$ and $\phi = 0.01$ correspond to the cases where $N_b = 0.1\, k = 12$ and $N_b = 0.01k = 1.2$ or, equivalently, $B_{\max} = 0.1\, k\, b = 12\, b$ and $B_{\max} = 0.01\, k\, b = 1.2\, b$, respectively. The values for the storage efficiency are chosen to be fractions of the form $z/(z+1)$, $z = 1, \ldots, 7$, such that the first point of each of the corresponding curves is associated with the single-parity $(z, z+1)$-erasure code, and the second point of each of the corresponding curves is associated with the double-parity $(2z, 2z+2)$-erasure code.

For all values of ϕ considered, we observe that the MTTDL increases as the storage efficiency s_{eff} decreases. This is because, for a given m, decreasing s_{eff} implies decreasing l, which in turn implies increasing the parity symbols $m - l$ and consequently improving the MTTDL. Furthermore, for a given storage efficiency, s_{eff}, the MTTDL decreases by orders of magnitude as the maximum network rebuild bandwidth decreases. We now proceed to identify the optimal codeword length, m^*, that maximizes the MTTDL for a given bandwidth constraint and storage efficiency. The optimal codeword length is dictated by two opposing effects on reliability. On the one hand, larger values of m imply that codewords can tolerate more device failures, but on the other hand, they result in a higher exposure degree to failure as each of the codewords is spread across a larger number of devices. In Figure 6, the optimal values, m^*, are indicated by the circles, and the corresponding codeword lengths are indicated by the vertical dotted lines. By comparing Figures 6(a), (b), and (c), we deduce that as ϕ decreases, so do the optimal codeword lengths. For example, in the case of $s_{\text{eff}} = 3/4$ and $\phi = 1$, the maximum MTTDL value of $4{\times}10^{78}$ is obtained when $m = m^* = 92$. However, in the case of $\phi = 0.1$, the maximum MTTDL value of $6{\times}10^{57}$ is obtained for $m^* = 84$. The reason for the reduction of the optimal codeword length is due to the fact that for a given value of s_{eff} and as m increases, so does $\tilde{r}$, which, according to Remark 3, results in a smaller reliability reduction factor. Thus, the reliability reduction factor corresponding to $m = 92$ is smaller than the one corresponding to $m = 84$, which in turn causes the MTTDL for $m = 92$ to no longer be optimal as it becomes smaller than the one for $m = 84$. Note that for $m = 84$ and $s_{\text{eff}} = 3/4$, from (1) and (3), it follows that $l = 63$ and $\tilde{r} - 1 = 21$. From (21), and given



(a) $\phi = 1$ (b) $\phi = 0.1$ (c) $\phi = 0.01$ (d) $\phi = 0.001$

Figure 6. Normalized MTTDL vs. codeword length for $s_{\text{eff}} = 1/2, 2/3, 3/4, 4/5, 5/6, 6/7,$ and $7/8$; $n = k = 120$, $\lambda/\mu = 0.001$.



(a) $\phi = 1$ (b) $\phi = 0.1$ (c) $\phi = 0.01$ (d) $\phi = 0.001$

Figure 7. Normalized EAFDL vs. codeword length for $s_{\text{eff}} = 1/2, 2/3, 3/4, 4/5, 5/6, 6/7,$ and $7/8$; $n = k = 120$, $\lambda/\mu = 0.001$.

that $u \le \tilde{r} - 1 = 21 \ll k = 120$, such that $\phi/(1 - u/k) \approx \phi$, it now follows that $\theta \approx \phi^{\tilde{r}-1} = 0.1^{21} = 10^{-21}$, which implies that the reliability is reduced by 21 orders of magnitude. In the cases of $\phi = 0.01$ and $\phi = 0.001$, the maximum MTTDL values of 6×10^{37} and 8×10^{19} are obtained for $m^* = 76$ and $m^* = 68$, respectively.

The combined effect of the network rebuild bandwidth constraint and the system efficiency on the normalized $\text{EAFDL}^{\text{declus}}/\lambda$ measure is obtained by (31) and (33), and shown in Figure 7 as a function of the codeword length. We observe that the EAFDL increases as the storage efficiency s_{eff} decreases. Furthermore, for a given storage efficiency, s_{eff}, the EAFDL increases by orders of magnitude as the maximum network rebuild bandwidth decreases. Similarly to the case of MTTDL, by comparing Figures 7(a), (b), and (c), we observe that as ϕ decreases, so do the optimal codeword lengths. For example, in the case of $s_{\text{eff}} = 3/4$ and $\phi = 1$, the minimum EAFDL value of 4×10^{-84} is obtained when $m = m^* = 88$. However, in the case of $\phi = 0.1$, the minimum EAFDL value of 9×10^{-64} is obtained for $m^* = 80$, which implies that the reliability is reduced by 20 orders of magnitude. In the cases of $\phi = 0.01$ and $\phi = 0.001$, the minimum EAFDL values of 2×10^{-44} and 6×10^{-27} are obtained for $m^* = 72$ and $m^* = 64$, respectively. By comparing Figures 6 and 7, we deduce that in general the optimal codeword lengths m^*_{MTTDL} (for MTTDL) and m^*_{EAFDL} (for EAFDL) are similar.

Reducing B_{max} or, equivalently, ϕ, affects the optimal codeword length as follows.

Proposition 1: For any storage efficiency s_{eff}, and for both reliability metrics, the optimal codeword length m^* decreases as ϕ decreases.

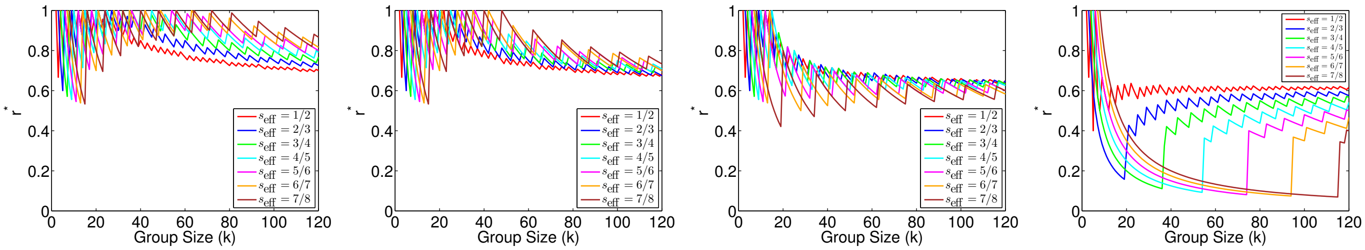
Proof: Consider two bandwidth constraint factors ϕ_1 and ϕ_2 with $\phi_1 > \phi_2$. Let m_1^* and m_2^* be the corresponding optimal codeword lengths for the MTTDL metric. We shall now show that $m_1^* \ge m_2^*$.

As m_1^* is the optimal codeword length for ϕ_1, it holds that $\text{MTTDL}(\phi_1, m) \le \text{MTTDL}(\phi_1, m_1^*)$ for all $m \ge m_1^*$. Also, from (1) and (3), it holds that $\tilde{r} = (1 - s_{\text{eff}})\, m + 1$, which implies that as m increases, so does $\tilde{r}$. From (15), it follows that $\theta^{(2)}/\theta^{(1)} = \prod_{u=1}^{\tilde{r}-1} \frac{b_u(\phi_2)}{b_u(\phi_1)}$, which, owing to the fact that $b_u(\phi_2) \le b_u(\phi_1)\ \forall u$, decreases as $\tilde{r}$ or, equivalently, m increases. Consequently, $\theta_m^{(2)}/\theta_m^{(1)} \le \theta_{m_1^*}^{(2)}/\theta_{m_1^*}^{(1)}$ for all $m \ge m_1^*$. Also, from (14), it follows that $\text{MTTDL}(\phi_2,m)/\text{MTTDL}(\phi_1,m) = \theta_m^{(2)}/\theta_m^{(1)}$ for all values of m. From the preceding, it follows that $\text{MTTDL}(\phi_2,m)/\text{MTTDL}(\phi_1,m) = \theta_m^{(2)}/\theta_m^{(1)} \le \theta_{m_1^*}^{(2)}/\theta_{m_1^*}^{(1)} = \text{MTTDL}(\phi_2,m_1^*)/\text{MTTDL}(\phi_1,m_1^*) \le \text{MTTDL}(\phi_2,m_1^*)/\text{MTTDL}(\phi_1,m)$ for all $m \ge m_1^*$. Thus, $\text{MTTDL}(\phi_2,m) \le \text{MTTDL}(\phi_2,m_1^*)$ for all $m \ge m_1^*$, which in turn implies that $m_2^* \le m_1^*$. The proof for EAFDL is similar to that for MTTDL and is therefore omitted. ■

From (22) and (23), it follows that the optimal codeword length depends on k and ϕ, but not on the storage system size, n. To investigate the behavior of the optimal codeword length, m^*, as the group size, k, increases, we proceed by considering the normalized optimal codeword length r^*, namely, the ratio of m^* to k:

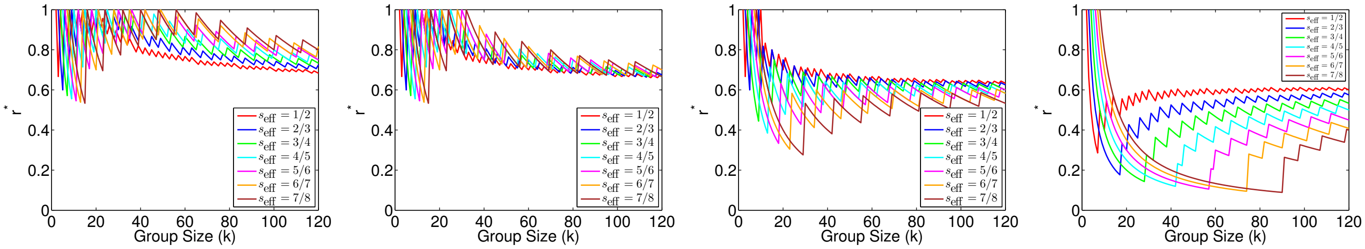
$$r^* \triangleq \frac{m^*}{k}\ . \tag{34}$$

The r^* values for the MTTDL and EAFDL metrics are shown in Figures 8 and 9, respectively, for various storage efficiencies. According to Proposition 1, for any storage efficiency s_{eff} and for any given group size k, the optimal codeword lengths and, consequently, the r^* values decrease as ϕ decreases. Also, when the bandwidth constraint factor ϕ is small, the r^* values first decrease and then gradually increase as k increases. The initial decrease is due to the fact that the optimal codeword length m^* remains fixed and equal to $z + 1$, which is the minimum possible codeword length for the storage efficiency fractions $z/(z + 1)$, $z = 1, \ldots, 7$. For example, in the case of $s_{\text{eff}} = 7/8$ and $\phi = 0.001$, $m^* = 8$ for $k < 115$ in the case of MTTDL, or for $k < 90$ in the case of EAFDL, as



(a) $\phi = 1$ (b) $\phi = 0.1$ (c) $\phi = 0.01$ (d) $\phi = 0.001$

Figure 8. r^* for MTTDL vs. group size for $s_{\text{eff}} = 1/2, 2/3, 3/4, 4/5, 5/6, 6/7,$ and $7/8$; $\lambda/\mu = 0.001$.



(a) $\phi = 1$ (b) $\phi = 0.1$ (c) $\phi = 0.01$ (d) $\phi = 0.001$

Figure 9. r^* for EAFDL vs. group size for $s_{\text{eff}} = 1/2, 2/3, 3/4, 4/5, 5/6, 6/7,$ and $7/8$; $\lambda/\mu = 0.001$.

shown in Figures 8(d) and 9(d), respectively. However, it can be proved that as k increases further, the r^* values for MTTDL and EAFDL approach a common value that depends only on the storage efficiency, s_{eff}, but not on the bandwidth constraint factor, ϕ, and are in the interval $[e^{-1/2} = 0.606, 0.648]$.

V. Discussion

Although erasure coding schemes provide a high data reliability at a high storage efficiency, the rebuild process involves I/O operations and network transfers that increase the consumption of device and network bandwidth. In particular, large MDS codes pose a challenge on the usage of network resources given that a lost symbol is recovered via an (m, l) erasure code through the transfer of a large number of l symbols from l surviving devices over the network. Consequently, recovering large amounts of data results in additional traffic over increased time periods, which has an impact on the latency of the foreground workload and therefore affects system performance. This issue, also known as the *repair bandwidth problem*, has prompted the development of alternative erasure coding schemes that aim at reducing the amount of data transferred over the storage network during reconstruction (see [35][36] and references therein). They can, however, result in higher amounts of data being read from the surviving devices and therefore in longer rebuild times. The effect of these methods on system reliability is beyond the scope of this paper and is a subject of further investigation.

The analytical findings of this work are relevant for the case of large data centers employing erasure coding where the excessive rebuild traffic competes with the huge amount of traffic generated by the frequent access of a large number of storage devices. To ensure a desired performance level, the network bandwidth devoted to the repair traffic needs to be contained. For small values of ϕ and k, a small codeword length should be selected, as discussed in Section IV. For large values of k, the codeword length should still be kept relatively small for performance reasons. This is in agreement with the practical values given in [36] for the various parameters considered. In particular, to keep the storage overhead low, the storage efficiency should be chosen in the range of 0.66 to 0.75.

VI. Conclusions

Data storage systems use erasure coding schemes to recover lost data and enhance system reliability. Network rebuild bandwidth constraints, however, may degrade reliability. A general methodology was applied for deriving the Mean Time to Data Loss (MTTDL) and the Expected Annual Fraction of Data Loss (EAFDL) reliability metrics analytically. Closed-form expressions capturing the effect of a network rebuild bandwidth constraint were obtained for the symmetric, clustered and declustered data placement schemes. We established that the reliability of storage systems is adversely affected by the network rebuild bandwidth constraints. The declustered placement scheme was found to offer superior reliability in terms of both metrics. An investigation of the reliability achieved by this scheme under various codeword configurations was subsequently conducted. The results obtained demonstrated that both metrics are optimized by similar codeword lengths. For large storage systems that use a declustered placement scheme, the optimized codeword lengths are about 60% of the storage system size, independently of the network rebuild bandwidth constraints. The analytical reliability expressions derived can be used to identify redundancy and recovery schemes, as well as data placement configurations that can achieve high reliability. The results obtained can also be used to adapt the data placement schemes when the available network rebuild bandwidth or the number of devices in the system changes so that the system maintains a high level of reliability.

Extending the methodology developed to derive the reliability of erasure coded systems under bandwidth constraints for arbitrary rebuild time distributions and in the presence of unrecoverable latent errors is a subject of further investigation. Also, owing to the parallelism of the rebuild process, the model considered yields very small rebuild times for large system sizes. Taking into account the fact that the rebuild times cannot be smaller than the actual failure detection times requires a more sophisticated modeling effort, which is also part of future work.

REFERENCES

[1] D. A. Patterson, G. Gibson, and R. H. Katz, "A case for redundant arrays of inexpensive disks (RAID)," in Proceedings of the ACM SIGMOD International Conference on Management of Data, Jun. 1988, pp. 109–116.

[2] P. M. Chen, E. K. Lee, G. A. Gibson, R. H. Katz, and D. A. Patterson, "RAID: High-performance, reliable secondary storage," ACM Comput. Surv., vol. 26, no. 2, Jun. 1994, pp. 145–185.

[3] M. Malhotra and K. S. Trivedi, "Reliability analysis of redundant arrays of inexpensive disks," J. Parallel Distrib. Comput., vol. 17, Jan. 1993, pp. 146–151.

[4] W. A. Burkhard and J. Menon, "Disk array storage system reliability," in Proceedings of the 23rd International Symposium on Fault-Tolerant Computing, Jun. 1993, pp. 432–441.

[5] K. S. Trivedi, Probabilistic and Statistics with Reliability, Queueing and Computer Science Applications, 2nd ed. New York: Wiley, 2002.

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

[7] T. J. E. Schwarz, Q. Xin, E. L. Miller, D. D. E. Long, A. Hospodor, and S. Ng, "Disk scrubbing in large archival storage systems," in Proceedings of the 12th Annual IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Oct. 2004, pp. 409–418.

[8] Q. Lian, W. Chen, and Z. Zhang, "On the impact of replica placement to the reliability of distributed brick storage systems," in Proc. 25th IEEE International Conference on Distributed Computing Systems (ICDCS), Jun. 2005, pp. 187–196.

[9] S. Ramabhadran and J. Pasquale, "Analysis of long-running replicated systems," in Proc. 25th IEEE International Conference on Computer Communications (INFOCOM), Apr. 2006, pp. 1–9.

[10] B. Eckart, X. Chen, X. He, and S. L. Scott, "Failure prediction models for proactive fault tolerance within storage systems," in Proceedings of the 16th Annual IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Sep. 2008, pp. 1–8.

[11] A. Thomasian and M. Blaum, "Higher reliability redundant disk arrays: Organization, operation, and coding," ACM Trans. Storage, vol. 5, no. 3, Nov. 2009, pp. 1–59.

[12] K. Rao, J. L. Hafner, and R. A. Golding, "Reliability for networked storage nodes," IEEE Trans. Dependable Secure Comput., vol. 8, no. 3, May 2011, pp. 404–418.

[13] I. Iliadis, R. Haas, X.-Y. Hu, and E. Eleftheriou, "Disk scrubbing versus intradisk redundancy for RAID storage systems," ACM Trans. Storage, vol. 7, no. 2, Jul. 2011, pp. 1–42.

[14] V. Venkatesan, I. Iliadis, C. Fragouli, and R. Urbanke, "Reliability of clustered vs. declustered replica placement in data storage systems," in Proceedings of the 19th Annual IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Jul. 2011, pp. 307–317.

[15] V. Venkatesan, I. Iliadis, and R. Haas, "Reliability of data storage systems under network rebuild bandwidth constraints," in Proceedings of the 20th Annual IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Aug. 2012, pp. 189–197.

[16] V. Venkatesan and I. Iliadis, "A general reliability model for data storage systems," in Proceedings of the 9th International Conference on Quantitative Evaluation of Systems (QEST), Sep. 2012, pp. 209–219.

[17] J.-F. Pâris, T. J. E. Schwarz, A. Amer, and D. D. E. Long, "Highly reliable two-dimensional RAID arrays for archival storage," in Proceedings of the 31st IEEE International Performance Computing and Communications Conference (IPCCC), Dec. 2012, pp. 324–331.

[18] V. Venkatesan and I. Iliadis, "Effect of codeword placement on the reliability of erasure coded data storage systems," in Proceedings of the 10th International Conference on Quantitative Evaluation of Systems (QEST), Sep. 2013, pp. 241–257.

[19] I. Iliadis and V. Venkatesan, "An efficient method for reliability evaluation of data storage systems," in Proceedings of the 8th International Conference on Communication Theory, Reliability, and Quality of Service (CTRQ), Apr. 2015, pp. 6–12.

[20] ——, "Most probable paths to data loss: An efficient method for reliability evaluation of data storage systems," Int'l J. Adv. Syst. Measur., vol. 8, no. 3&4, Dec. 2015, pp. 178–200.

[21] S. Caron, F. Giroire, D. Mazauric, J. Monteiro, and S. Pérennes, "P2P storage systems: Study of different placement policies," Peer-to-Peer Networking and Applications, Mar. 2013, pp. 1–17.

[22] I. Iliadis and V. Venkatesan, "Expected annual fraction of data loss as a metric for data storage reliability," in Proceedings of the 22nd Annual IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), Sep. 2014, pp. 375–384.

[23] ——, "Reliability assessment of erasure coded systems," in Proceedings of the 10th International Conference on Communication Theory, Reliability, and Quality of Service (CTRQ), Apr. 2017, pp. 41–50.

[24] ——, "Reliability evaluation of erasure coded systems," Int'l J. Adv. Telecommun., vol. 10, no. 3&4, Dec. 2017, pp. 118–144.

[25] J. G. Elerath and J. Schindler, "Beyond MTTDL: A closed-form RAID 6 reliability equation," ACM Trans. Storage, vol. 10, no. 2, Mar. 2014, pp. 1–21.

[26] I. Iliadis and V. Venkatesan, "Rebuttal to 'Beyond MTTDL: A closed-form RAID-6 reliability equation'," ACM Trans. Storage, vol. 11, no. 2, Mar. 2015, pp. 1–10.

[27] "Amazon Simple Storage Service." [Online]. Available: http://aws.amazon.com/s3/ [retrieved: November 2017]

[28] D. Borthakur et al., "Apache Hadoop goes realtime at Facebook," in Proceedings of the ACM SIGMOD International Conference on Management of Data, Jun. 2011, pp. 1071–1080.

[29] R. J. Chansler, "Data availability and durability with the Hadoop Distributed File System," ;login: The USENIX Association Newsletter, vol. 37, no. 1, 2013, pp. 16–22.

[30] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The Hadoop Distributed File System," in Proceedings of the 26th IEEE Symposium on Mass Storage Systems and Technologies (MSST), May 2010, pp. 1–10.

[31] D. Ford et al., "Availability in globally distributed storage systems," in Proceedings of the 9th USENIX Symposium on Operating Systems Design and Implementation (OSDI), Oct. 2010, pp. 61–74.

[32] C. Huang et al., "Erasure coding in Windows Azure Storage," in Proceedings of the USENIX Annual Technical Conference (ATC), Jun. 2012, pp. 15–26.

[33] S. Muralidhar et al., "f4: Facebook's Warm BLOB Storage System," in Proceedings of the 11th USENIX Symposium on Operating Systems Design and Implementation (OSDI), Oct. 2014, pp. 383–397.

[34] "IBM Cloud Object Storage." [Online]. Available: www.ibm.com/cloud-computing/products/storage/object-storage/how-it-works/ [retrieved: November 2017]

[35] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network coding for distributed storage," Proc. IEEE, vol. 99, no. 3, Mar. 2011, pp. 476–489.

[36] M. Zhang, S. Han, and P. P. C. Lee, "A simulation analysis of reliability in erasure-coded data centers," in Proceedings of the 36th IEEE Symposium on Reliable Distributed Systems (SRDS), Sep. 2017, pp. 144–153.