Index Assignment for Multiple Description Repair in Distributed Storage Systems

Dževdan Kapetanović, Symeon Chatzinotas, Björn Ottersten
Interdisciplinary Center for Security, Reliability and Trust (SnT)
University of Luxembourg, Luxembourg
Email: dzevdan.kapetanovic@ericsson.com, {symeon.chatzinotas, bjorn.ottersten}@uni.lu

Abstract—Distributed storage systems have been receiving increasing attention lately due to the developments in cloud and grid computing. Furthermore, a major part of the stored information comprises of multimedia, whose content can be communicated even with a lossy (non-perfect) reconstruction. In this context, Multiple Description Lattice Quantizers (MDLQ) can be employed to encode such sources for distributed storage and store them across distributed nodes. Their inherent properties yield that having access to all nodes gives perfect reconstruction of the source, while the reconstruction quality decreases gracefully with fewer available nodes. If a set of nodes fails, lossy repair techniques could be applied to reconstruct the failed nodes from the available ones. This problem has mostly been studied with the lossless (perfect) reconstruction assumption. In this work, a general model, Multiple Description Lattice Quantizer with Repairs (MDLQR), is introduced that encompasses the lossy repair problem for distributed storage applications. New performance measures and repair techniques are introduced for MDLQR, and a non-trivial identity is derived, which is related to other results in the literature. This enables us to find the optimal encoder for a certain repair technique used in the MDLQR. Furthermore, simulation results are used to evaluate the performance of the different repair techniques.

I. INTRODUCTION

Distributed storage refers to representing an information source with data units and storing them across a number of (probably geographically distributed) nodes. The objective is to improve reliability, since the information source can be reconstructed using the data units of a subset of nodes. This area has gained importance due to the recent developments in cloud networks and related applications [1], [2].

Starting from this point, we can distinguish two different cases depending on the type of information source reconstruction: *lossless* and *lossy* reconstruction. In the former case, the information source has to be perfectly reconstructed in order to be useful, e.g. documents, software etc. (n, k) Maximum Distance Separable (MDS) codes (e.g. [3], [4]) can be used in order to ensure that any k data units can perfectly reconstruct a source represented by n data units. However, this means that obtaining access to more than k data units provides no improvement in reconstruction, while obtaining access to less than k data units conveys no information at all. This phenomenon is also known as the "cliff" effect (see Fig. 1). In the latter case, lossy reconstruction can still convey some useful message, e.g. music, photos, video. This is due to the way humans perceive media files, meaning that the message can still be conveyed from imperfect representations
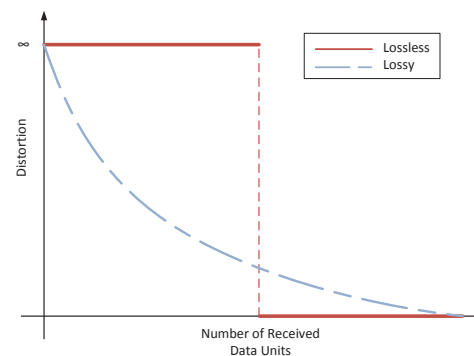

Fig. 1. Conceptional plot of distortion level with number of received units for lossless and lossy source reconstruction.

of the source. Furthermore, an estimate of the source can be acquired even when less that k units are available, while the reconstruction quality keeps improving even when more than k units are received. In this direction, Multiple Description (MD) codes [5], [6] and quantizers [7] can be employed in order to ensure that the reconstruction quality is an increasing function of the available nodes k.

Independently of the coding technique, a common problem in distributed storage systems is how to repair the nodes that are bound to fail from time to time. When a node fails, the information it stored is lost forever but it may be possible to perfectly or approximately recover it using the redundancy information stored across the distributed storage system. In lossless reconstruction, this problem has been tackled in the literature taking also into account the repair network traffic [8] and exploiting the interference alignment concept inspired by communication networks [9], [10].

However, there is limited work in the literature about repairing nodes in distributed multimedia storage systems which can afford lossy reconstruction. To the authors' knowledge, this was first investigated in [11] from an experimental point of view. This paper builds on [11], presenting an analytical model of the lossy repair problem and introducing three different repair techniques. For one of the techniques, we derive a result that shows how to construct the optimal encoder. More specifically, in section II, the system model is described and the repair problem is formally defined. In section III, the different repair techniques are presented and a repair distortion measure is introduced. Section IV minimizes the repair distortion for one of the repair techniques. Finally, section V presents numerical results of the source and repair distortions of the different
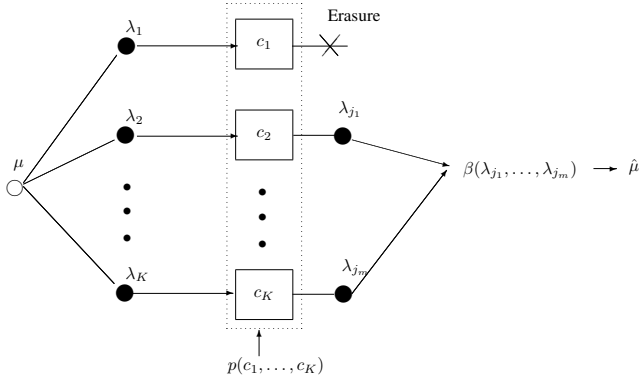
Fig. 2. Scalar Multiple Description Lattice Quantizer (MDLQ).

techniques, and section VI concludes the paper.

II. SYSTEM MODEL

In this section, we start by presenting a mathematical model of the scalar multiple description lattice quantizer (MDLQ) system. This system encompasses the distributed storage problem of interest. Afterwards, we introduce the MDLQ with repairs (MDLQR) system, which is the focus of this work.

A. Scalar MDLQ system

A scalar MDLQ system is depicted in Figure 2. A source outputs real-valued symbols $\{X_j\}_{j=0}^{\infty}$ belonging to some alphabet $\mathcal{X}$. The source symbol X is quantized to the closest element $\mu \in \mathcal{A}_c$, where $\mathcal{A}_c$ is a *central codebook* which is a discrete (typically one-dimensional) lattice. The volume of a Voronoi cell in $\mathcal{A}_c$ is denoted as ν. Next, μ is mapped onto a K-dimensional vector $(\lambda_1(\mu), \ldots, \lambda_K(\mu))$ by an encoding function $\lambda : \mathcal{A}_c \to \mathcal{A}_1 \times \ldots \times \mathcal{A}_K$, where $\mathcal{A}_k \subset \mathcal{A}_c$ is the *side codebook* for λ_k, $1 \leq k \leq K$ (when no confusion arises, we will omit denoting the dependency of $\lambda_k(\mu)$ on μ)[1][2]. The side codebooks are also lattices, typically similar to the central codebook [12], of index $I = [\mathcal{A}_c : \mathcal{A}_k]$, $1 \leq k \leq K$. The element λ_j, $j = 1, \ldots, K$, also known as a *description*, is coded by R_j bits (e.g. by entropy coding) and transmitted across the j:th channel that carries R_j bits at most. Let $h(X)$ denote the differential entropy of the source. The rate R_c of the central codebook is approximately [12] $R_c \approx h(X) - \log_2(\nu)$. The rate R_j of the j:th side codebook is $R_j \approx h(X) - \log_2(I\nu) = R_c - \log_2(I)$. At the decoder, the information across channel j is either received error free or completely discarded if errors occur. If the j:th channel is free of errors, then the corresponding *channel indicator* (CI) variable $c_j = 1$, otherwise $c_j = 0$. A joint CI probability function (CIPF) $p(c_1, \ldots, c_K)$ represents the error probability of the channels. The decoder observes $m \leq K$ descriptions $\lambda_{j_1}, \ldots, \lambda_{j_m}, 1 \leq j_1 \leq \ldots \leq j_m \leq K$ and applies a decoding function $\beta : \mathcal{A}_{j_1} \times \ldots \times \mathcal{A}_{j_m} \to \mathcal{A}_c$ to obtain an estimate $\hat{\mu} = \beta(\lambda_{j_1}, \ldots, \lambda_{j_m})$ of the sent symbol. The main practical problem is to design the encoding and decoding functions, such that the rate constraints of the channels are met, in order

[1] In distributed storage applications, each element λ_j, $1 \leq j \leq K$ is stored in a node.

[2] The encoding is also refered to as *index assignment* in the literature.

to minimize the average distortion $D = E\{\|\mu - \hat{\mu}\|^2\}$, where henceforth $\|.\|$ denotes the Frobenius norm. The expectation is taken over the central codebook and the CI variables c_j, $j = 1, \ldots, K$.

In the case of independent CIs, the model in Figure 2 reduces to the classical symmetric, scalar MDLQ problem (henceforth, we implicitly assume scalar systems, and thus omit writing scalar). This problem has been studied from an achievability point of view in [13]–[16], where the goal is to determine the minimum possible distortion for each set of constraint rates. In [12], [17], [18], the same problem is studied from a practical point of view, where the aim is to design practical encoding and decoding functions achieving low distortions.

In contrast, for a distributed storage application, dependency between CIs is common. For example, if only one channel at a time fails, then all joint erasure probabilities equal 0. Thus, for distributed storage applications, it is of interest to specify the joint distribution of the CIs, rather than individual probabilities.

B. Multiple Description Lattice Quantizer with Repairs

In Figure 3, the setup is similar to Figure 2, but with the main difference being that we have consecutive MDLQ systems, with the possibility of repairing the lost descriptions in each step [11]. We call this system the *multiple description lattice quantizer with repairs* (MDLQR). If no intermediate repairs are allowed, MDLQR is equivalent to MDLQ. However, if repairs are allowed at each step, performance gains are possible and the problem is different. For example, assume that the original descriptions $\lambda_1, \ldots, \lambda_K$ are completely repaired at some step in the MDLQR. Then, with higher probability, the destination will receive more descriptions than without repairing, which produces a lower source distortion. Note that for MDLQR, the CIs and the CIPFs are now depending on the step j in the system. Hence, at each step, a new probability function determines the error behavior.

The MDLQR is characterized by three different functions: an encoding function λ as in the MDLQ, a repair function $\alpha : \mathcal{A}_{j_1} \times \ldots \times \mathcal{A}_{j_m} \to \mathcal{A}_{j_{m+1}} \times \ldots \times \mathcal{A}_{j_K}$, and a final decoding function $\beta : \mathcal{A}_1 \times \ldots \times \mathcal{A}_K \to \mathcal{A}_c$. Note that the repair function has the original side codebooks as input and output as well. Let $\lambda_j^{(k)} \in \mathcal{A}_j$ denote the *j:th description at the k:th step* of the MDLQR, $1 \leq k \leq N$, with $\lambda_j^{(0)} \triangleq \lambda_j$ being the output of the encoding function λ. In general, after several steps, we have $\lambda_j^{(k)} \neq \lambda_j^{(0)}$. As for the MDLQ, the main goal of the MDLQR is to minimize the final source distortion

$$D_s = E\{\mu - \beta(\lambda_1^{(N)}, \ldots, \lambda_K^{(N)})\}. \tag{1}$$

The expectation is across the central codebook and the CIs. Note that D_s depends on the encoding function λ, the repair function α and the decoding function β.

III. ENCODING, REPAIR AND DECODING FUNCTIONS

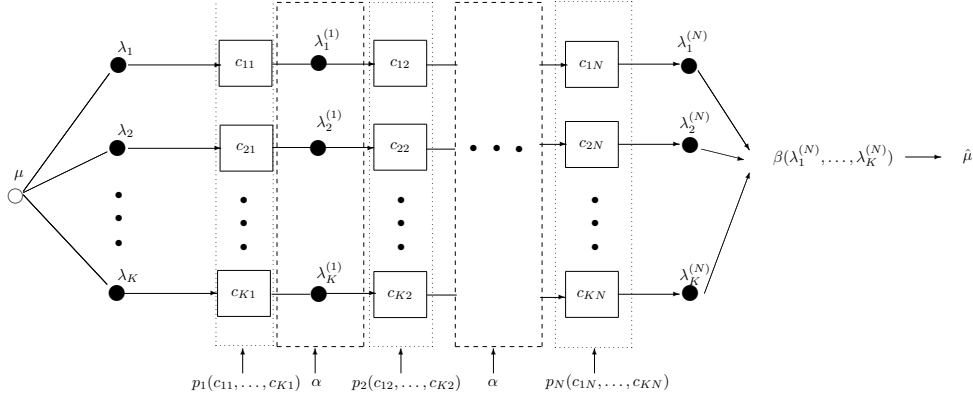The encoding function is assumed to be injective, both for the MDLQ system and the MDLQR, in order to uniquely

Fig. 3. Scalar Multiple Description Lattice Quantizer with Repairs (MDLQR).

Name	Description
	At step k:
CDrep	For $m < l \leq K$, the lost descriptions are recovered as $$\lambda_{j_l}^{(k)} = \alpha_{j_l} = Q_{j_l}\left\{\frac{\sum_{s=1}^{m}\lambda_{j_s}^{(k)}}{m}\right\}.$$
CDSrep	For $m < l \leq K$, the lost descriptions are recovered as $$\lambda_{j_l}^{(k)} = \alpha_{j_l} = Q_{j_l}\left(Q_c\left\{\frac{\sum_{s=1}^{m}\lambda_{j_s}^{(k)}}{m}\right\}\right).$$
CDSSrep	First, the source symbol $\hat{\mu} = Q_c\left\{\frac{\sum_{s=1}^{m}\lambda_{j_s}^{(k)}}{m}\right\}$ is estimated. Next, the closest source symbol μ to $\hat{\mu}$ that has the descriptions that match $\lambda_{j_1}^{(k)}, \ldots, \lambda_{j_m}^{(k)}$ is chosen. The lost descriptions are then $\lambda_{j_{m+1}}^{(k)}(\mu), \ldots, \lambda_{j_K}^{(k)}(\mu)$.

TABLE I
REPAIR TECHNIQUES INVESTIGATED IN THIS PAPER.

identify the source from its descriptions. Beside this requirement, the encoding function is constructed in a way that minimizes the source distortion. However, this construction clearly depends on the specific repair and decoding functions that are used. The decoding function, β, used in this work is the *common decoding* (CD) method [12] [17] [18]

$$\beta(\lambda_{j_1}^{(N)}, \ldots, \lambda_{j_m}^{(N)}) = \frac{\sum_{k=1}^{m}\lambda_{j_k}^{(N)}}{m}. \quad (2)$$

The main advantage of using the common decoding method is its simplicity and low complexity.

We will investigate three different repair functions in this work, and characterize the optimal λ function for one of them. The repair functions are summarized in Table I. In the table, $Q_j\{.\}$ denotes the quantization (mapping to the closest element) to the j:th side codebook, with $Q_c\{.\}$ denoting quantization to $\mathcal{A}_c$. CDSSrep has the highest computational complexity, since a search in the central codebook (which is denser than the side codebooks) is necessary for recovering the lost descriptions. The quantization operation has a negligible complexity, and thus CDrep and CDSrep have roughly the same complexity, with CDrep having $K-1$ additional quantization operations compared to CDSrep. On the other hand, CDSrep needs an index assignment lookup table, which requires additional memory.

In order to measure and compare the performance of different repair functions in the first step of the MDLQR, we introduce the *first order repair distortion*, which is defined as

$$
\begin{aligned}
D_r^{(1)} &\triangleq E\{\|(\lambda_{j_{m+1}}, \ldots, \lambda_{j_K}) - \alpha(\lambda_{j_1}, \ldots, \lambda_{j_m})\|^2\} \\
&= E\{\|(\lambda_1, \ldots, \lambda_K) - (\lambda_1^{(1)}, \ldots, \lambda_K^{(1)})\|^2\}. \quad (3)
\end{aligned}
$$

The expectation is over the central codebook and the CIs in the first step. Hence, (3) measures the average distortion between the original symbols and the symbols after the first step in the MDLQR. Minimizing this distortion results in an index assignment that is robust to the first step in the MDLQR. This minimization will be performed for the CDrep repair function. Let $\mathcal{L}^{K,m}$ denote the set of m-tuples taken from $\{1, 2, \ldots, K\}$. As an example, $l = (l_1\,l_2\,l_3) = (1\,3\,4)$ is an m-tuple in $\mathcal{L}^{K,3}$ for $K \geq 4$. Furthermore, let $\Pr\{c_{l_1 1}, \ldots, c_{l_k 1}\}$ denote the probability of error free reception across channels $l_1, \ldots, l_k$ and erroneous reception across the rest (henceforth, $\Pr\{.\}$ is the probability operator). With this notation, a more explicit expression for (3) is

$$
\begin{aligned}
D_r^{(1)} &= \sum_{\mu \in \mathcal{A}_c} \sum_{l \in \mathcal{L}^{K,m}} \Pr\{\mu\}\Pr\{c_{l_1 1}, \ldots, c_{l_k 1}\} \\
&\sum_{n=m+1}^{K} (\lambda_{l_n}(\mu) - \alpha_n(\lambda_{l_1}(\mu), \ldots, \lambda_{l_m}(\mu)))^2. \quad (4)
\end{aligned}
$$

IV. MINIMIZING THE FIRST ORDER REPAIR DISTORTION FOR CDREP

The goal in this section is to find an encoding function that minimizes the first order repair distortion in (3) (from now on, for convenience, refered to as the *repair distortion*) for CDrep. In this work, we assume that the source is uniformly distributed. Beside being a practical assumption, it is also a natural one if no prior information about the source is at hand. Moreover, we assume that $\Pr\{c_{l_1 1}, \ldots, c_{l_k 1}\}$ has the same value for any realization of $l_1, \ldots, l_m$ (i.e., we assume a symmetrical system) with the additional constraint that all other events (i.e., fewer or more than m erasures) have probability 0. These assumptions are also implicitly used in [18]. In order to obtain a mathematically tractable problem, we will discard the quantization operation $Q_j\{.\}$, which then

results in $\alpha = \beta$, i.e., the repair function is the common decoding method. Under the above assumptions, (3) becomes proportional to

$$D_r^{(1)} \sim \tilde{D}_r^{(1)} = \sum_{\mu \in \mathcal{A}_c} \sum_{l \in \mathcal{L}^{K,m}} \sum_{j=m+1}^{K} \left(\lambda_{l_j} - \frac{\sum_{k=1}^{m} \lambda_{l_k}}{m} \right)^2. \quad (5)$$

As in [12] [18], define

$$\bar{\lambda} \triangleq \frac{\sum_{j=1}^{K} \lambda_j}{K}.$$

We now prove
Theorem 1.

$$\begin{aligned}
\tilde{D}_r^{(1)} &= \frac{K(m+1)\binom{K-2}{m-1}}{m^2} \sum_{\mu \in \mathcal{A}_c} \sum_{j=1}^{K} (\lambda_j - \bar{\lambda})^2 \\
&= \frac{(m+1)\binom{K-2}{m-1}}{m^2} \sum_{\mu \in \mathcal{A}_c} \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} (\lambda_i - \lambda_j)^2.
\end{aligned}$$

Proof. See Appendix. $\square$

Geometrically, the first equality in Theorem 1 implies that minimizing the repair distortion for CDrep amounts to finding K points $\lambda_1, \ldots, \lambda_K$ that are closest to their centroid. The second equality in Theorem 1 shows that this is equivalent to finding K points whose sum of pairwise squared distances (SPSD) is minimal. Theorem 1 also shows that the repair distortion is proportional to the SSD costs defined in [18] and the SPSD term in Theorem 3.1 in [12]. Hence, the optimal index assignment presented in those papers can be applied to minimize the repair distortion. In [18], it is shown that if the side codebooks are translated lattices, the index assignment minimizing the SSD costs is obtained by finding shortest vectors in translated A_{K-1} lattices. In this case, the minimum value of $\tilde{D}_r^{(1)}$ for CDrep can be expressed through the theta series of the A_{K-1} lattice. The theta series can be calculated only up to 3 dimensions, while it is unknown for higher dimensions [18].

V. NUMERICAL RESULTS

In this section, we numerically evaluate the source and repair distortions for the three repair techniques in Table I.

As side codebooks, we use translated lattices as in [18]. The i:th codebook is $\mathcal{A}_i = \{K\epsilon z_i + (2i - K + 1)\epsilon/2 \; : \; z_i \in \mathbb{Z}\}$, where $\mathbb{Z}$ is the set of integers and ϵ is a scaling factor. The reference lattice $\mathcal{A}_r = \{\frac{\omega_1 + \ldots + \omega_K}{K} \; : \; \omega_i \in \mathcal{A}_i, 1 \leq j \leq K\} = \{\epsilon z : z \in \mathbb{Z}\}$ arising from the centroids of the different description is dense for the translated lattices, compared to the side codebooks which are of lower density. Hence, as a result, a good reconstruction of the source and the descriptions is achieved from the CD method when using translated lattices [18]. Based on $\mathcal{A}_r$, the central lattice is constructed as $\mathcal{A}_c = \{\frac{\epsilon}{M}z + \frac{\epsilon}{2M}\text{mod}(M+1, 2) \; : \; z \in \mathbb{Z}\}$, where $\text{mod}(a, b)$ means a modulo b. This definition of $\mathcal{A}_c$ ensures that there are exactly M central lattice points within the Voronoi region of $\mathcal{A}_r$. It follows that the index I equals $I = KM$, since
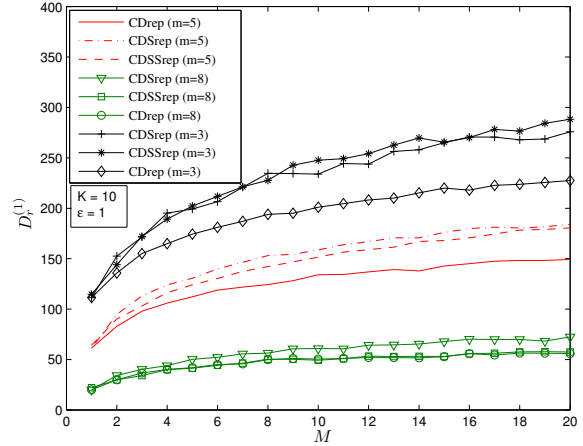

Fig. 4. $D_r^{(1)}$ versus M for $K = 10$ descriptions and scaling factor $\epsilon = 1$. The repair distortion increases with M, since longer vectors in the A_{K-1} lattice are used as descriptions. Moreover, $D_r^{(1)}$ decreases with increasing m. CDrep is the most efficient repairing method, followed by CDSSrep and CDSrep.

there are KM central lattice points within the Voronoi cell of a side codebook. Hence, the central codebook rate is $R_c \approx h(X) - \log_2(\nu)$, where $\nu = \epsilon/M$. The rate R_j of the j:th side codebook is $R_j \approx h(X) - \log_2(I\nu) = R_c - \log_2(KM)$. Hence, increasing M increases the gap in rate between the side descriptions and the central lattice points. Theorem 1 and results in [18] show that for translated lattices, the optimal λ that minimizes the repair distortion for CDrep is constructed by translating the M shortest vectors from the A_{K-1} lattice. We use this index assignment for all the repairing techniques and the CD method.

To evaluate the repair distortion, a central lattice point is chosen uniformly, and m out of its K descriptions are also chosen uniformly – the other $K - m$ descriptions are reconstructed with the repair techniques in Table I. This corresponds to a probability distribution where $\Pr\{c_{l_1 1}, \ldots, c_{l_k 1}\} = 1/\binom{K}{m}$ for any realization of $l_1, \ldots, l_m$. This procedure is repeated 300 000 times in order to obtain an estimate of $D_r^{(1)}$. In Figure 4, $D_r^{(1)}$ is evaluated for different values of M. As seen, the distortion increases slightly with M since longer vectors in the A_{K-1} lattice are used as descriptions, which increases the repair distortion. Moreover, the repair distortion decreases when more descriptions are received, which is expected. The most efficient repairing method is CDrep, followed by CDSSrep and CDSrep. Note that when few descriptions are received, CDrep is significantly better than the other two, while CDSSrep comes close to CDrep when more descriptions are received. Since CDSSrep is of slightly higher complexity than CDrep (a search is necessary), CDrep is to prefer.

For the source distortion, in each simulation instance, a central lattice point is chosen uniformly, and its descriptions are sent through the N cascaded channels in Figure 3. It is assumed that each step in the system produces $K - m$ erasures, where m is predetermined. It is further assumed that all the CIPFs are the same as the one described above for the repair distortion. When repairing, at each step in the repair
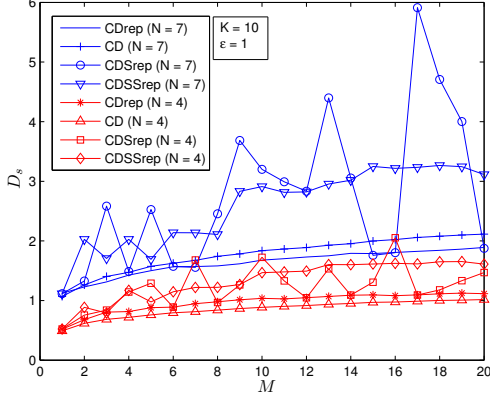
Fig. 5. D_s versus M for $m = 1$, $K = 10$ descriptions and scaling factor $\epsilon = 1$. The source distortion increases with M, since longer vectors in the A_{K-1} lattice are used as descriptions. Furthermore, more steps in the MDLQR makes CDrep the most effective technique, while CD gives lowest distortion for a few steps.

system, a repair technique from Table I is applied. At the receiver, K descriptions are obtained, on which the common decoding method is applied in order to estimate the original central lattice point. If no repairing is performed, the common decoding method is applied on the descriptions that have not been erased.

In Figure 5, the source distortion is evaluated for $m = 1$. Without repairing, more steps (larger N) in the MDLQR worsens the performance of the CD method, since fewer descriptions are received at the destination. However, with repairs, CDrep performs better than CD. Hence, in case of many erasures, repairing has the possibility to recover the source with lower distortion than the well-known CD method. For a few number of erasures ($N = 4$), CD produces a lower distortion than any repair technique, since faulty repairing introduces additional distortion which dominates for a small number of erasures. Note that CDSrep and CDSSrep have discontinuous jumps in distortion for different M values. The reason is that with increasing M, new descriptions are included such that when being repaired with search methods CDSrep and CDSSrep, give rise to vectors that are far from or sometimes close to the source symbol after the final common decoding reconstruction. This results in increased or decreased distortion, which thus fluctuates with M. Figure 6 shows the source distortion for $m = 2$. Same conclusions can be drawn as for $m = 1$.

VI. CONCLUSIONS AND FUTURE WORK

In this work, we presented an analytical model of the lossy repair problem in distributed storage. Different repairing techniques were defined, and a repair distortion resulting from performing a single repair is introduced. The optimal index assignment minimizing the repair distortion is derived for the CDrep repair method. Through numerical simulations, it is shown that CDrep is the best repairing method when using translated lattices and the index assignment that is optimal for CDrep. Moreover, for the final source distortion, repairing erasures with CDrep gives better performance than the common
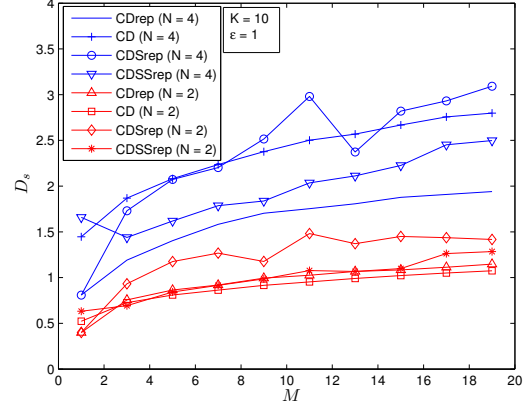

Fig. 6. D_s versus M for $m = 2$, $K = 10$ descriptions and scaling factor $\epsilon = 1$. The source distortion increases with M, since longer vectors in the A_{K-1} lattice are used as descriptions. Furthermore, more steps in the MDLQR makes CDrep the most effective technique, while CD gives lowest distortion for a few steps.

decoding method when several consecutive erasures occur. On the other hand, for few erasures, CD performs better than all repair techniques. Future work will attempt to find the optimal index assignment minimizing the repair distortion for CDrep for more than one step in the MDLQR.

VII. APPENDIX

Here we prove Theorem 1. Let $\mathcal{L}_{\sim j}^{K,m}$ denote the set of m-tuples taken from $\{1, 2, \ldots, K\}$ that do not contain index j. For example, $\mathcal{L}_{\sim 1}^{4,2} = \{(2\,3), (2\,4), (3\,4)\}$. $\tilde{D}_r^{(1)}$ in (5) can now be rewritten as

$$
\begin{aligned}
\tilde{D}_r^{(1)} &= \sum_{\mu \in \mathcal{A}_c} \sum_{j=m+1}^{K} \sum_{l \in \mathcal{L}^{K,m}} \left(\lambda_{l_j} - \frac{\sum_{k=1}^{m} \lambda_{l_k}}{m} \right)^2 \\
&= \sum_{\mu \in \mathcal{A}_c} \sum_{j=1}^{K} \sum_{l \in \mathcal{L}_{\sim j}^{K,m}} \left(\lambda_j - \frac{\sum_{k=1}^{m} \lambda_{l_k}}{m} \right)^2. \quad (6)
\end{aligned}
$$

The second equality follows by noting that l_j takes on all the values $1, \ldots, K$ in the first identity. We start by expanding the third inner sum in (6) for a fixed j:

$$
\sum_{l \in \mathcal{L}_{\sim j}^{K,m}} \left(\lambda_j - \frac{\sum_{k=1}^{m} \lambda_{l_k}}{m} \right)^2 = \binom{K-1}{m} \lambda_j^2
$$

$$
- \frac{2\lambda_j}{m} \binom{K-2}{m-1} \sum_{\substack{k=1 \\ k \neq j}} \lambda_k + \frac{1}{m^2} \sum_{l \in \mathcal{L}_{\sim j}^{K,m}} \left(\sum_{k=1}^{m} \lambda_{l_k} \right)^2 \quad (7)
$$

$$
= \left(\binom{K-1}{m} + \frac{2}{m} \binom{K-2}{m-1} \right) \lambda_j^2 - \frac{2\lambda_j}{m} \binom{K-2}{m-1} \sum_{k=1}^{K} \lambda_k \quad (8)
$$

$$
+ \frac{1}{m^2} \sum_{l \in \mathcal{L}_{\sim j}^{K,m}} \left(\sum_{k=1}^{m} \lambda_{l_k} \right)^2, \quad (9)
$$

where the last equality follows by adding to and subtracting from the first sum in (7) the term λ_j. When summing the

above expression over j, the double sum in (9) becomes

$$\sum_{j=1}^{K} \sum_{l \in \mathcal{L}_{\sim j}^{K,m}} \left(\sum_{k=1}^{m} \lambda_{l_k} \right)^2 = (K-1)\binom{K-2}{m-1} \sum_{k=1}^{K} \lambda_k^2 \quad (10)$$

$$+ (K-2)\binom{K-3}{m-2} \sum_{\substack{k,n=1 \\ k \neq n}}^{K} \lambda_k \lambda_n$$

$$= \left((K-1)\binom{K-2}{m-1} - (K-2)\binom{K-3}{m-2} \right) \sum_{j=1}^{K} \lambda_j^2$$

$$+ (K-2)\binom{K-3}{m-2} \left(\sum_{j=1}^{K} \lambda_j \right)^2. \quad (11)$$

The first equality holds by noting that a fixed λ_i^2 in the left hand double sum of (10) occurs exactly $(K-1)\binom{K-2}{m-1}$ times, since it appears in $K-1$ inner sums and in each inner sum it occurs exactly $\binom{K-2}{m-1}$ times. Similarly, a cross product $\lambda_i \lambda_k$ appears in $K-2$ inner sums and in each inner sum it occurs $\binom{K-3}{m-2}$ times. For the second equality, we complete the squares by expressing the sum of cross products as a sum of squares. By summing the expressions in (8) and (9) over j, using the formula in (11), and collecting terms, we get

$$\tilde{D}_r^{(1)} = \sum_{\mu \in \mathcal{A}_c} \frac{1}{m^2} \left(m^2 \binom{K-1}{m} + 2m \binom{K-2}{m-1} \right)$$

$$+ (K-1)\binom{K-2}{m-1} - (K-2)\binom{K-3}{m-2} \right) \sum_{j=1}^{K} \lambda_j^2$$

$$+ \frac{1}{m^2} \left((K-2)\binom{K-3}{m-2} - 2m\binom{K-2}{m-1} \right) \left(\sum_{j=1}^{K} \lambda_j \right)^2. \quad (12)$$

By straightforward algebraic manipulations, it follows that

$$m^2\binom{K-1}{m} + 2m\binom{K-2}{m-1} + (K-1)\binom{K-2}{m-1}$$

$$- (K-2)\binom{K-3}{m-2} = K(m+1)\binom{K-2}{m-1}$$

and

$$(K-2)\binom{K-3}{m-2} - 2m\binom{K-2}{m-1} = -(m+1)\binom{K-2}{m-1}.$$

Hence, (12) equals

$$\tilde{D}_r^{(1)} = \frac{K(m+1)\binom{K-2}{m-1}}{m^2} \sum_{\mu \in \mathcal{A}_c} \left(\sum_{j=1}^{K} \lambda_j^2 - \frac{1}{K} \left(\sum_{j=1}^{K} \lambda_j \right)^2 \right). \quad (13)$$

Further, it holds that

$$\sum_{j=1}^{K} \lambda_j^2 - \frac{1}{K} \left(\sum_{j=1}^{K} \lambda_j \right)^2 = \sum_{j=1}^{K} \left(\lambda_j - \frac{1}{K} \sum_{k=1}^{K} \lambda_k \right)^2 \quad (14)$$

$$= \frac{1}{K} \sum_{i=1}^{K-1} \sum_{j=1}^{K} (\lambda_i - \lambda_j)^2 \quad (15)$$

Inserting (14) and (15) into (13) gives the desired identities in Theorem 1 and completes the proof.

REFERENCES

[1] W. Zhu, C. Luo, J. Wang, and S. Li, "Multimedia cloud computing," *Signal Processing Magazine, IEEE*, vol. 28, no. 3, pp. 59 –69, May 2011.

[2] A. Marinos and G. Briscoe, "Community cloud computing," in *Cloud Computing*, ser. Lecture Notes in Computer Science, M. Jaatun, G. Zhao, and C. Rong, Eds. Springer Berlin Heidelberg, 2009, vol. 5931, pp. 472–484.

[3] I. S. Reed and G. Solomon, "Polynomial Codes Over Certain Finite Fields," *Journal of the Society for Industrial and Applied Mathematics*, vol. 8, no. 2, pp. 300–304, 1960. [Online]. Available: *http://dx.doi.org/10.1137/0108018*

[4] M. O. Rabin, "Efficient dispersal of information for security, load balancing, and fault tolerance," *J. ACM*, vol. 36, no. 2, pp. 335–348, 1989. [Online]. Available: *http://doi.acm.org/10.1145/62044.62050*

[5] S. Pradhan, R. Puri, and K. Ramchandran, "n-channel symmetric multiple descriptions - part i: (n, k) source-channel erasure codes," *Information Theory, IEEE Transactions on*, vol. 50, no. 1, pp. 47 – 61, Jan. 2004.

[6] R. Puri, S. Pradhan, and K. Ramchandran, "n-channel symmetric multiple descriptions-part ii:an achievable rate-distortion region," *Information Theory, IEEE Transactions on*, vol. 51, no. 4, pp. 1377 – 1392, April 2005.

[7] G. Zhang, J. Klejsa, and W. Kleijn, "Optimal index assignment for multiple description scalar quantization with translated lattice codebooks," *Signal Processing, IEEE Transactions on*, vol. 60, no. 8, pp. 4444 –4451, Aug. 2012.

[8] A. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," *Proceedings of the IEEE*, vol. 99, no. 3, pp. 476 –489, March 2011.

[9] S. Chatzinotas and B. Ottersten, "Cognitive interference alignment between small cells and a macrocell," in *Telecommunications (ICT), 2012 19th International Conference on*, April 2012, pp. 1 –6.

[10] ——, "Interference mitigation techniques for clustered multicell joint decoding systems," *EURASIP Journal on Wireless Communications and Networking, Multicell Cooperation for Next Generation Communication Systems Series*, vol. 132, 2011.

[11] S. Chatzinotas, "Repairing multiple description quantizers in distributed storage systems," in *IEEE International Conference on Communications, ICC 2013*, 2013.

[12] J. Ostergaard, J. Jensen, and R. Heusdens, "n-channel entropy-constrained multiple-description lattice vector quantization," *Information Theory, IEEE Transactions on*, vol. 52, no. 5, pp. 1956 – 1973, May 2006.

[13] A. A. E. Gamal and T. Cover., "Achievable rates for multiple descriptions," *Information Theory, IEEE Transactions on*, vol. 28, pp. 851–857, Nov. 1982.

[14] Z. Zhang and T. Berger, "New results in binary multiple descriptions," *Information Theory, IEEE Transactions on*, vol. 33, no. 4, pp. 502–521, July 1987.

[15] L. Ozarow, "On a source-coding problem with two channels and three receivers," *Bell Syst. Tech. J.*, vol. 59, no. 10, pp. 1909–1921, Dec. 1980.

[16] R. Zamir, "Gaussian codes and shannon bounds for multiple descriptions," *Information Theory, IEEE Transactions on*, vol. 45, pp. 2629–2635, Nov. 1999.

[17] J. Ostergaard and R. Zamir, "Multiple-description coding by dithered delta-sigma quantization," *Information Theory, IEEE Transactions on*, vol. 55, no. 10, pp. 4661 –4675, Oct. 2009.

[18] G. Zhang, J. Ostergaard, J. Klejsa, and W. Kleijn, "High-rate analysis of symmetric l-channel multiple description coding," *Communications, IEEE Transactions on*, vol. 59, no. 7, pp. 1846 –1856, July 2011.