Distributed Storage Allocation Problems

Derek Leong, Alexandros G. Dimakis, Tracey Ho
Department of Electrical Engineering
California Institute of Technology
Pasadena, California 91125, USA
{*derekleong, adim, tho*}*@caltech.edu*

Abstract — **We investigate the problem of using several storage nodes to store a data object, subject to an aggregate storage budget or redundancy constraint. It is challenging to find the optimal allocation that maximizes the probability of successful recovery by the data collector because of the large space of possible symmetric and nonsymmetric allocations, and the nonconvexity of the problem. For the special case of probability-1 recovery, we show that the optimal allocation that minimizes the required budget is symmetric. We further explore several storage allocation and access models, and determine the optimal symmetric allocation in the high-probability regime for a case of interest. Based on our experimental investigation, we make a general conjecture about a phase transition on the optimal allocation.**

I. Introduction

We consider the problem of using n storage nodes and choosing the most reliable way of allocating storage among them, subject to an aggregate storage budget or redundancy constraint. A source has a data object of unit size and is allowed to use any coding scheme to store x_1 amount of data in the first storage node, x_2 in the second, and so on, as illustrated in Fig. 1. The only constraint is that the total amount of data stored over all the storage nodes does not exceed some given storage budget T:

$$\sum_{i=1}^{n} x_i \leq T.$$

At some time after the creation of this encoded storage, a data collector accesses the data stored in a subset $\mathbf{r}$ of the storage nodes and tries to recover the original data object. The problem of finding a *good storage allocation* (i.e. determining $\{x_i\}_{i=1}^n$) and that of creating an actual code that realizes this allocation decouple: if a good coding scheme is used, successful recovery is possible when the amount of data accessed by the data collector is at least the data object size:

$$\mathbb{P}[\text{successful recovery}] = \mathbb{P}\left[\sum_{i\in\mathbf{r}} x_i \geq 1\right].$$

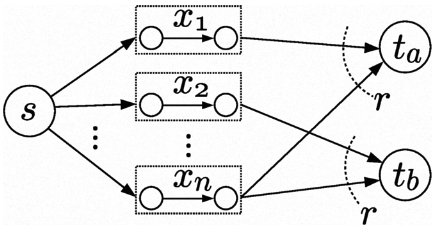

Fig. 1: Example of a distributed storage allocation problem. The source s has a data object of unit size which is to be stored in a distributed manner over n storage nodes, each storing x_i amount of data. The data collector is allowed to access only r storage nodes; t_a and t_b are two such realizations of the data collector.

This can be seen by formulating the distributed storage problem as a network flow problem in which the source wishes to multicast to the data collectors [1, 2]; network coding allows us to achieve a multicast rate equal to the smallest max-flow among all the data collectors. The storage of random linear combinations of data packets over a sufficiently large field, for example, would allow such recovery with high probability [3, 4].

We are interested in determining the allocation described by $\{x_i\}_{i=1}^n$ that maximizes the probability of successful recovery subject to the storage budget constraint. Note that in our setup, the communication links between the nodes are able to support the amount of data on the respective storage nodes; in other scenarios, the link capacities rather than the storage capacities might limit the storage budget T.

The intended application dictates a *failure model* which induces a probability measure on the subsets of storage nodes that can be accessed by a data collector. Prior work on distributed storage for sensor networks [5–8] assumes that a random subset of fixed cardinality is selected and that all x_i are equal. In this paper, we address only the allocation problem; it will be interesting to construct sparse codes and efficient decoding algorithms, such as those introduced in [5–8], that work well under various allocations.

A related natural model is to assume that each storage node is accessed by the data collector independently with some constant probability p, which means that $|\mathbf{r}|$ is a binomial random variable. This *deterministic allocation* with *probabilistic access* problem is an open problem discussed by several people at UC Berkeley [9], and it is known that the optimal allocation that maximizes the probability of successful recovery can be quite compli-

This material is based upon work under a subcontract #069153 issued by BAE Systems National Security Solutions, Inc. and supported by the Defense Advanced Research Projects Agency (DARPA) and the Space and Naval Warfare System Center (SPAWARSYSCEN), San Diego under Contract No. N66001-08-C-2013.

978-1-4244-4724-4/09/$25.00 © 2009 IEEE

cated in general — for instance, the following counterexample (originally from [9]) shows that symmetric allocations (i.e. all nonzero x_i are equal) can be suboptimal: given $n = 5$, $p = 0.9$, and $T = \frac{12}{5}$, the nonsymmetric allocation $\left(\frac{3}{5}, \frac{3}{5}, \frac{2}{5}, \frac{2}{5}, \frac{2}{5}\right)$ yields a success probability of 0.99711, which is strictly greater than the corresponding probabilities for the five symmetric allocations, of which $\left(\frac{3}{5}, \frac{3}{5}, \frac{3}{5}, \frac{3}{5}, 0\right)$ achieves the highest success probability of 0.9963. The problem appears nontrivial even if we restrict our optimization over symmetric allocations.

Another variation of the model allows the data collector to access a random r-subset of storage nodes selected uniformly from the collection of all possible r-subsets, where $r = |\mathbf{r}|$ is a constant. For this case of *deterministic allocation* with *deterministic access*, we have not found any nonsymmetric allocation that outperforms the optimal symmetric allocation; we conjecture that for any budget T, there always exists a symmetric allocation that produces the optimal success probability. For example, given $n = 5$, $r = 2$, and $T = \frac{12}{5}$, the maximum success probability of 0.7 can be attained by both the nonsymmetric allocation $\left(\frac{3}{5}, \frac{3}{5}, \frac{2}{5}, \frac{2}{5}, \frac{2}{5}\right)$ and the symmetric allocation $\left(\frac{6}{5}, \frac{6}{5}, 0, 0, 0\right)$. Again, the optimal allocation is not obvious even if we consider only symmetric allocations: we observe numerically that for most choices of (n, r, T), the optimal symmetric allocation either concentrates the budget over a minimal number of nodes, or spreads it maximally; an example of an exception is $(n, r, T) = (15, 3, 4.6)$ for which the optimal number of nodes to use, 9, is neither of the extremes. As discussed later, we observe that this complication does not occur for *probabilistic allocations* which seem to have a direct phase transition from minimal to maximal spread.

Our Contribution: Our first result is that if the data collector accesses a random r-subset of storage nodes, and recovery must be successful with probability 1, i.e. *all* r-subsets must allow successful recovery, then the symmetric allocation over all n storage nodes:

$$x_i = \frac{1}{r}, \qquad i = 1, 2, \ldots, n,$$

minimizes the required budget T.

Our second result involves *symmetric* probabilistic storage allocations: we flip a coin for each storage node and with probability p decide to use it to store a fixed amount of data. For probabilistic allocations, the total storage used $\sum_{i=1}^{n} x_i$ is a random variable which we require to be no greater than the budget T in expectation. Suppose that each nonempty storage node stores $\frac{1}{\ell}$ amount of data, and the data collector is allowed to access any r-subset of storage nodes. We show that if the budget T is large enough to allow a probability of successful recovery above a given threshold, then the choice of $\ell = r$ maximizes the success probability for that given budget.

II. Allocations for Probability-1 Recovery

Suppose the data collector accesses each r-subset $\mathbf{r}$ with probability $p_{\mathbf{r}}$, where $\mathbf{r}$ belongs to the collection $\mathcal{R}$ of all $\binom{n}{r}$ r-subsets of storage nodes. We seek the allocation $\{x_i\}_{i=1}^{n}$ that minimizes the budget T, among all allocations that achieve a probability of successful recovery of at least P:

$$\text{minimize} \quad T = \sum_{i=1}^{n} x_i$$

$$\text{subject to}$$

$$\sum_{\mathbf{r} \in \mathcal{R}} p_{\mathbf{r}} \, \mathbf{I}_{\{\sum_{i \in \mathbf{r}} x_i \geq 1\}} \geq P$$

$$x_i \geq 0, \; i = 1, 2, \ldots, n$$

In the special case of probability-1 recovery (i.e. $P = 1$), the problem reduces to a simple linear program with $\binom{n}{r}$ r-subset constraints of the form $\sum_{i \in \mathbf{r}} x_i \geq 1$, assuming $p_{\mathbf{r}} > 0$ for all $\mathbf{r} \in \mathcal{R}$. We proceed to find a (sorted) allocation

$$0 \leq x_1 \leq x_2 \leq \cdots \leq x_n$$

that minimizes the total storage budget $T = \sum_{i=1}^{n} x_i$, so that the amount of data stored on any r-subset of storage nodes, $r < n$, is at least the data object size:

$$\sum_{i \in \mathbf{r}} x_i \geq 1 \qquad \forall \, r\text{-subset } \mathbf{r} \subset \{1, 2, \ldots, n\}.$$

It is not hard to solve this optimization problem and show that the symmetric allocation over all n storage nodes is optimal:

Theorem 1. *Choosing $x_i = \frac{1}{r}$, $i = 1, 2, \ldots, n$, minimizes $T = \sum_{i=1}^{n} x_i$, subject to $\sum_{i \in \mathbf{r}} x_i \geq 1 \; \forall \, r$-subset $\mathbf{r} \subset \{1, 2, \ldots, n\}$, $x_i \geq 0$, $i = 1, 2, \ldots, n$.*

Intuitively, this result is not surprising — a *symmetric* optimal allocation makes sense because all r-subsets are equally weighted; to minimize the budget, each node would need to store only $\frac{1}{r}$ amount of data to ensure successful recovery.

Proof. We first use a simple argument to show that an optimal allocation $\{x_i\}_{i=1}^{n}$ must be *symmetric* (i.e. all nonzero x_i are equal). We subsequently optimize over all symmetric allocations and show that the one that uses all the nodes minimizes the budget.

Lemma 1. *For any feasible* nonsymmetric *allocation $\{x_i\}_{i=1}^{n}$, we can construct a feasible* symmetric *allocation $\{x_i'\}_{i=1}^{n}$ such that $\sum_{i=1}^{n} x_i' < \sum_{i=1}^{n} x_i$.*

Proof of Lemma 1. Since $x_1, x_2, \ldots, x_r$ are the r smallest elements in the allocation, we have $1 \leq \sum_{i=1}^{r} x_i \leq \sum_{i \in \mathbf{r}} x_i \; \forall \, r$-subset $\mathbf{r} \subset \{1, 2, \ldots, n\}$. We construct a *symmetric* allocation $\{x_i'\}_{i=1}^{n}$ whose nonzero elements are equal to the mean taken over only the nonzero elements in $\{x_i\}_{i=1}^r$:

$$x_i' = \begin{cases} 0 & \text{if } x_i = 0, \\ \bar{x} & \text{otherwise,} \end{cases} \qquad \bar{x} = \frac{\sum_{i=1}^r x_i}{\sum_{i=1}^r \mathbf{I}_{\{x_i>0\}}}.$$

Note that there are less than r zero elements in $\{x_i\}_{i=1}^n$, otherwise there is at least one r-subset of elements whose sum is zero which would mean that the allocation is infeasible. It follows that the denominator $\sum_{i=1}^r \mathbf{I}_{\{x_i>0\}} \geq 1$. The constructed allocation $\{x_i'\}_{i=1}^n$ is indeed feasible since

$$0 \leq x_1' \leq x_2' \leq \cdots \leq x_n', \text{ and}$$

$$1 \leq \sum_{i=1}^r x_i = \sum_{i=1}^r x_i' \leq \sum_{i\in\mathbf{r}} x_i' \quad \forall r\text{-subset } \mathbf{r} \subset \{1,2,\ldots,n\},$$

observing again that $x_1', x_2', \ldots, x_r'$ are the r smallest elements in the allocation. Furthermore, $\sum_{i=r+1}^n x_i' < \sum_{i=r+1}^n x_i$, since $x_i' = \bar{x} \leq x_r \leq x_i \quad \forall i > r$, and either (i) $x_r = \bar{x}$ in which case all nonzero elements x_i, $i \leq r$, are equal to $\bar{x} \Rightarrow$ there exists an x_i, $i > r$, such that $x_i > \bar{x} = x_i'$ because the allocation is nonsymmetric, or (ii) $x_r > \bar{x}$ in which case $x_i' = \bar{x} < x_r \leq x_i \quad \forall i > r$. Therefore

$$\begin{aligned}\sum_{i=1}^n x_i' = \sum_{i=1}^r x_i' + \sum_{i=r+1}^n x_i' &= \sum_{i=1}^r x_i + \sum_{i=r+1}^n x_i' \\ &< \sum_{i=1}^r x_i + \sum_{i=r+1}^n x_i = \sum_{i=1}^n x_i. \blacksquare\end{aligned}$$

All feasible nonsymmetric allocations are therefore strictly suboptimal; it suffices to find the optimal symmetric allocation. Consider a feasible symmetric allocation $\{x_i\}_{i=1}^n$ containing θ zero elements, where $\theta < r$ (otherwise there is at least one r-subset of elements whose sum is zero). Let

$$x_{\theta+1} = \cdots = x_n = x > 0.$$

Since the allocation is feasible, the sum of the r smallest elements must be at least 1, i.e. $\sum_{i=1}^r x_i = (r-\theta)x \geq 1 \Rightarrow x \geq \frac{1}{r-\theta}$. It follows that $\sum_{i=1}^n x_i = (n-\theta)x$ is minimized for a given θ when $x = \frac{1}{r-\theta}$. Writing the corresponding sum in terms of θ gives

$$T(\theta) = \sum_{i=1}^n x_i = \frac{n-\theta}{r-\theta}, \text{ with } T'(\theta) = \frac{n-r}{(r-\theta)^2} > 0.$$

Therefore $T(\theta)$ is minimized when $\theta = 0$, which gives $x = \frac{1}{r}$ and $\sum_{i=1}^n x_i = \frac{n}{r}$. $\blacksquare$

III. Symmetric Allocations

We turn our attention to *symmetric* allocations under a more general framework that allows for deterministic *vs* probabilistic storage allocation and access by the data collector. Determining the conditions under which symmetric allocations are optimal remains an open problem in general. We introduce *probabilistic* allocations where each storage node is used with probability $\frac{s}{n}$. This simplifies the analysis by keeping the probability of accessing a used storage node constant, and has the same effect as accessing storage nodes with replacement. As such, the probabilistic allocation serves as an approximation for the deterministic allocation that uses s storage nodes when the number of nodes accessed $r \ll s$. We adopt the following notation:

n	total number of storage nodes, used and unused
S	total number of storage nodes *used*
$\frac{1}{\ell}$	amount of data in each of the S storage nodes *used*, where $\ell \in \mathbb{Z}^+$
T	expected total storage budget, i.e. $T = \mathbb{E}[S]\frac{1}{\ell}$
R	number of storage nodes, used and unused, accessed by the data collector
Z	number of *used* storage nodes accessed by the data collector
P	probability of successful recovery $\mathbb{P}[Z \geq \ell]$

Let $\mathrm{Bin}(n,p)$ denote the binomial probability distribution with n trials and success probability p. For brevity, we denote sums of the binomial tail as

$$\mathbb{P}[\mathrm{Bin}(n,p) \geq k] = \sum_{i=k}^n \binom{n}{i} p^i (1-p)^{n-i}.$$

In general, we are interested in the optimal symmetric allocation specified by design parameters S, ℓ, T, and R that maximizes the success probability P, subject to some given constraint (e.g. choosing an optimal ℓ for a given constant number of storage nodes accessed R and total storage budget T). This problem can be posed under deterministic or probabilistic allocation and access models, as presented in Table 1.

A. Probabilistic Allocation with Deterministic Access

We will now study a specific model of allocation and access, namely case II in Table 1, which leads to a simple derivation of the optimal symmetric allocation in the high-probability regime. Storage nodes are used probabilistically, but the number of storage nodes accessed by the data collector is fixed:

$$S \sim \mathrm{Bin}\left(n, \frac{s}{n}\right), \text{ and } R = r,$$

where $s \in \mathbb{R}^+$ and $r \in \mathbb{Z}^+$ are constants. Clearly, the number of accessed nodes that have data, Z, is a binomial random variable and the probability of successful recovery is

$$P(r,s,\ell) = \mathbb{P}[Z \geq \ell] = \mathbb{P}\left[\mathrm{Bin}\left(r, \frac{s}{n}\right) \geq \ell\right].$$

Reparameterizing in terms of a fixed expected total storage budget $T = \mathbb{E}[S]\frac{1}{\ell} = \frac{s}{\ell}$ gives us

$$P(r,T,\ell) = \mathbb{P}\left[\mathrm{Bin}\left(r, \frac{T\ell}{n}\right) \geq \ell\right].$$

Table 1: Symmetric Allocations — Deterministic *vs* Probabilistic Storage Allocation and Access Models.

	Allocation	Access	Probability of Successful Recovery $P(r,s,\ell)=\mathbb{P}[Z\geq\ell]$
I	deterministic $S=s$ constant $s\in\mathbb{Z}^+$	deterministic $R=r$ constant $r\in\mathbb{Z}^+$	$\sum_{i=\ell}^{r}\mathbb{P}\left[Z=i\right]=\sum_{i=\ell}^{r}\frac{\binom{s}{i}\binom{n-s}{r-i}}{\binom{n}{r}}$
II	probabilistic $S\sim\text{Bin}\left(n,\frac{s}{n}\right)$ constant $s\in\mathbb{R}^+$	deterministic $R=r$ constant $r\in\mathbb{Z}^+$	$\mathbb{P}\left[\text{Bin}\left(r,\frac{s}{n}\right)\geq\ell\right]$
III	deterministic $S=s$ constant $s\in\mathbb{Z}^+$	probabilistic $R\sim\text{Bin}\left(n,\frac{r}{n}\right)$ constant $r\in\mathbb{R}^+$	$\mathbb{P}\left[\text{Bin}\left(s,\frac{r}{n}\right)\geq\ell\right]$
IV	probabilistic $S\sim\text{Bin}\left(n,\frac{s}{n}\right)$ constant $s\in\mathbb{R}^+$	probabilistic $R\sim\text{Bin}\left(n,\frac{r}{n}\right)$ constant $r\in\mathbb{R}^+$	$\mathbb{P}\left[\text{Bin}\left(n,\frac{s}{n}\cdot\frac{r}{n}\right)\geq\ell\right]$

We define $P(r,T,\ell)=1$ for $\frac{T\ell}{n}>1\Leftrightarrow T>\frac{n}{\ell}$, $\ell=1,2,\ldots,r$, which corresponds to the case of excess storage budget, i.e. all storage nodes are used. For a given number of accessed nodes r and expected budget T, we seek the optimal choice of ℓ (which together with T specifies the allocation) that maximizes $P(r,T,\ell)$. We now show that $\ell=r$ is optimal in the high-probability regime:

Theorem 2. *For any $r\geq 2$, and for any budget T large enough to support some symmetric allocation with success probability $P(r,T,\ell)>0.9$, the choice of $\ell=r$ is optimal among symmetric allocations, i.e. it maximizes $P(r,T,\ell)$ over all ℓ.*

Proof. We need only consider $\ell\in\{1,2,\ldots,r\}$ since $\ell>r$ corresponds to the case where accessing r used storage nodes would yield only $\frac{r}{\ell}<1$ amount of data which is insufficient for recovery. Observe that at $T=\frac{n}{r}$, the choice of $\ell=r$ gives success probability $P\left(r,T=\frac{n}{r},\ell=r\right)=1$; if we can upper-bound the success probabilities of the other choices of ℓ at $T=\frac{n}{r}$, then we have $\ell=r$ optimal whenever its success probability is at least that bound. We now proceed to find such a bound.

Lemma 2. *For any $r\geq 2$, if at $T=\frac{n}{r}$, the success probability*

$$P\left(r,T=\frac{n}{r},\ell\right)\leq\alpha,$$

for all $\ell\in\{1,2,\ldots,r-1\}$, where α is some threshold, then the choice of $\ell=r$ maximizes $P(r,T,\ell)$ over all ℓ for any

$$T\geq\frac{n}{r}\alpha^{\frac{1}{r}}.$$

Proof of Lemma 2. For $T\geq\frac{n}{r}$, the choice of $\ell=r$ is optimal because $P(r,T,\ell=r)=1$ is maximal. For other choices $\ell\in\{1,2,\ldots,r-1\}$, if we have $P\left(r,T=\frac{n}{r},\ell\right)\leq\alpha$, where α is some threshold, then it follows that $P(r,T,\ell)\leq\alpha$ for any $T\leq\frac{n}{r}$, since $P(r,T,\ell)$ is nondecreasing in T because

$$\frac{dP}{dT}=\binom{r}{\ell}\frac{\ell}{T}\left(\frac{T\ell}{n}\right)^{\ell}\left(1-\frac{T\ell}{n}\right)^{r-\ell}\geq 0,\qquad 0<\frac{T\ell}{n}\leq 1.$$

Thus, whenever

$$P(r,T,\ell=r)\geq\alpha\Leftrightarrow\left(\frac{Tr}{n}\right)^{r}\geq\alpha\Leftrightarrow T\geq\frac{n}{r}\alpha^{\frac{1}{r}},$$

the choice of $\ell=r$ is optimal. ∎

Lemma 3. *For any $r\geq 2$, and any $\ell\in\{1,2,\ldots,r-1\}$, we have the following upper bound on the success probability at $T=\frac{n}{r}$:*

$$P\left(r,T=\frac{n}{r},\ell\right)\leq\frac{1}{2}+\frac{\mathrm{e}^{\frac{1}{12r}}}{\sqrt{2\pi}}\sqrt{\frac{r}{r-1}}.$$

Proof of Lemma 3. At $T=\frac{n}{r}$, we have

$$P\left(r,T=\frac{n}{r},\ell\right)=\mathbb{P}\left[\text{Bin}\left(r,\frac{\ell}{r}\right)\geq\ell\right].$$

Since the mean of the binomial distribution $\text{Bin}\left(r,\frac{\ell}{r}\right)$ is ℓ which is an integer, its median coincides with the mean [10]. For any $r\geq 2$, and any $\ell\in\{1,2,\ldots,r-1\}$, the pmf evaluated at the median ℓ is

$$\begin{aligned}\mathbb{P}\left[\text{Bin}\left(r,\frac{\ell}{r}\right)=\ell\right]&=\binom{r}{\ell}\left(\frac{\ell}{r}\right)^{\ell}\left(1-\frac{\ell}{r}\right)^{r-\ell}\\&<\frac{\mathrm{e}^{\frac{1}{12r}}}{\sqrt{2\pi}}\sqrt{\frac{r}{\ell(r-\ell)}}\qquad(1)\\&\leq\max_{1\leq\ell\leq r-1}\frac{\mathrm{e}^{\frac{1}{12r}}}{\sqrt{2\pi}}\sqrt{\frac{r}{\ell(r-\ell)}}\\&=\frac{\mathrm{e}^{\frac{1}{12r}}\sqrt{r}}{\sqrt{2\pi}}\max_{1\leq\ell\leq r-1}\frac{1}{\sqrt{\ell(r-\ell)}}\\&=\frac{\mathrm{e}^{\frac{1}{12r}}}{\sqrt{2\pi}}\sqrt{\frac{r}{r-1}}\triangleq c_r.\qquad(2)\end{aligned}$$

Inequality (1) follows from the application of the following analytically convenient bound:

$$\binom{n}{k}<\frac{\mathrm{e}^{\frac{1}{12n}}}{\sqrt{2\pi}}\frac{k^{-k-1/2}\,n^{n+1/2}}{(n-k)^{n-k+1/2}},\qquad 1\leq k<n,$$

which is obtained by applying the upper and lower Stirling-based bounds (attributed to Feller, see e.g. [11]):

$$\sqrt{2\pi n}\left(\frac{n}{\mathrm{e}}\right)^n < n! < \sqrt{2\pi n}\left(\frac{n}{\mathrm{e}}\right)^n \mathrm{e}^{\frac{1}{12n}}, \qquad n \geq 1.$$

Equality (2) follows from the observation that the term in the square root $f(\ell) \triangleq \ell(r-\ell)$ is concave in ℓ. Therefore

$$\min_{1\leq\ell\leq r-1} f(\ell) = \min\{f(1), f(r-1)\} = \min\{r-1, r-1\}$$

$$= r-1 \Rightarrow \max_{1\leq\ell\leq r-1} \frac{1}{\sqrt{f(\ell)}} = \frac{1}{\sqrt{r-1}}.$$

By definition, the median m of a discrete random variable X satisfies

$$\mathbb{P}[X \leq m] = \mathbb{P}[X < m] + \mathbb{P}[X = m] \geq 1/2.$$

Using this inequality and $\mathbb{P}[X = m] \leq c_r$ gives

$$\mathbb{P}\left[\mathrm{Bin}\left(r, \frac{\ell}{r}\right) = \ell\right] \leq c_r \Rightarrow \mathbb{P}\left[\mathrm{Bin}\left(r, \frac{\ell}{r}\right) \geq \ell\right] \leq \frac{1}{2} + c_r.$$

■

Lemma 4. *For any $r \geq 2$, the choice of $\ell = r$ maximizes $P(r, T, \ell)$ over all ℓ for any*

$$T \geq \frac{n}{r}\left(\frac{1}{2} + \frac{\mathrm{e}^{\frac{1}{12r}}}{\sqrt{2\pi}}\sqrt{\frac{r}{r-1}}\right)^{\frac{1}{r}}.$$

Since $P(r, T, \ell)$ is nondecreasing in T, we can also restate this as follows: For any $r \geq 2$, at each T that for some ℓ supports a success probability

$$P(r, T, \ell) > \frac{1}{2} + \frac{\mathrm{e}^{\frac{1}{12r}}}{\sqrt{2\pi}}\sqrt{\frac{r}{r-1}} \triangleq p_r,$$

the choice of $\ell = r$ is optimal, i.e. it maximizes $P(r, T, \ell)$ over all ℓ.

Proof of Lemma 4. The lemma is a direct consequence of Lemmas 2 and 3. ■

Observe that p_r is decreasing in r because

$$\frac{dp_r}{dr} = -\frac{\mathrm{e}^{\frac{1}{12r}}\,(7r-1)\,(r-1)^{3/2}}{12\sqrt{2\pi}\,(r-1)^3\,r^{3/2}} < 0, \qquad r \geq 2.$$

In particular, we note that

$$p_r \to \frac{1}{2} + \frac{1}{\sqrt{2\pi}} \approx 0.8989, \qquad \text{as } r \to \infty.$$

Substituting $r = 221$ into Lemma 4 gives $p_r = 0.89999$, which means that for all $r \geq 221$, the choice of $\ell = r$ is optimal at each T that for some ℓ supports $P(r, T, \ell) > 0.9$. After numerically verifying that the same is true for $2 \leq r < 221$, we arrive at Theorem 2. ■

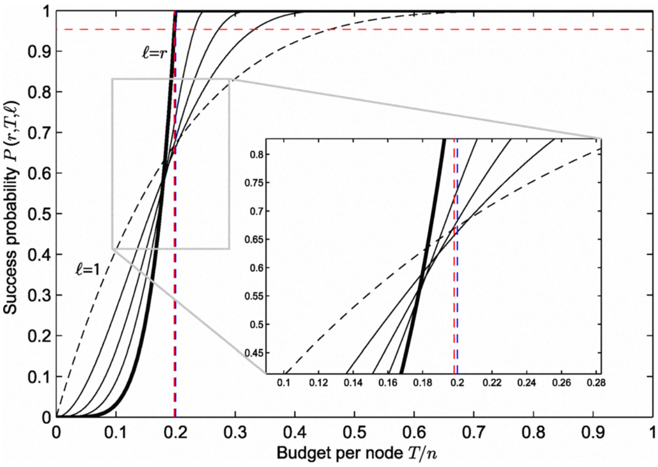

Fig. 2: Success probability $P(r, T, \ell)$ against storage budget per node $\frac{T}{n}$, for $r = 5$. The five curves correspond to different choices of $\ell = 1, 2, 3, 4, 5$; the dashed curve represents $\ell = 1$, and the bold curve represents $\ell = r$. The horizontal dashed line is $P(r, T, \ell) = p_r \approx 0.95$, and the vertical dashed lines are $\frac{T}{n} = \frac{1}{r}(p_r)^{\frac{1}{r}} \approx 0.198$ and $\frac{T}{n} = \frac{1}{r} = 0.2$.

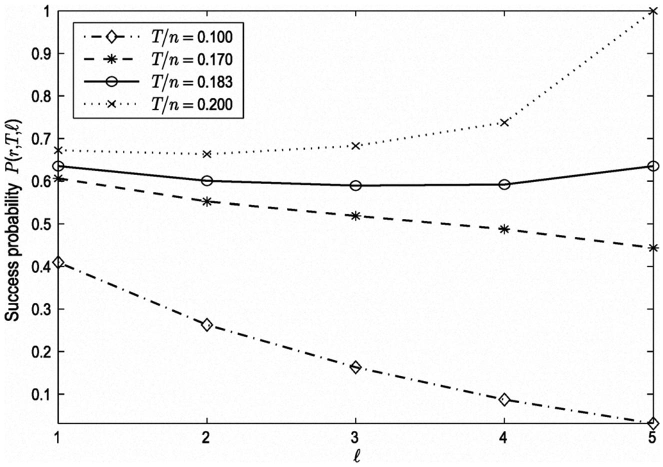

Fig. 3: Success probability $P(r, T, \ell)$ against ℓ, for $r = 5$. The four curves correspond to different storage budgets per node $\frac{T}{n}$. The solid curve corresponds to the critical budget where the phase transition occurs, i.e. $P(r, T, \ell = 1) = P(r, T, \ell = r)$.

We have also observed numerically that the claim holds even if we expand the range of budgets to include any T large enough to support some symmetric allocation with success probability $P(r, T, \ell) > 0.75$.

IV. Experimental Results

We numerically investigated the performance of the bounds derived in the preceding section for symmetric probabilistic allocation with deterministic access. Fig. 2 shows a typical plot of success probability $P(r, T, \ell)$ against the storage budget per node $\frac{T}{n}$, for $r = 5$. We observe that the choice of $\ell = 1$ is optimal when $\frac{T}{n} < 0.183$; otherwise $\ell = r$ is optimal. Fig. 3 shows the corresponding success probabilities $P(r, T, \ell)$ at each ℓ, given different storage budgets per node $\frac{T}{n}$. Increasing ℓ has different effects on the success probability P, depending on the budget. At high budgets, it monotonically increases P; at low budgets, it monotonically decreases P; at intermediate budgets, P becomes convex in ℓ.

Evidently, we can do much better than the derived bound in terms of the success probability (see Lemma 4): even for values of $P(r,T,\ell)$ as low as 0.64 for $r=5$, the choice of $\ell = r$ is already optimal. Fortunately, because of the steep slope of $P(r,T,\ell=r) = \left(\frac{T}{n}r\right)^r$, we do much better in terms of the bound on the expected total budget, i.e. choosing $\ell = r$ for any $T \geq \frac{n}{r}(p_r)^{\frac{1}{r}}$.

Our experiments motivate the following conjecture: for any r and T, the optimal choice of ℓ that maximizes success probability $P(r,T,\ell)$ is either $\ell = 1$ or $\ell = r$, and the switch from the optimality of $\ell = 1$ to that of $\ell = r$ occurs at the root of the equation $\left(\frac{Tr}{n}\right)^r = 1 - \left(1 - \frac{T}{n}\right)^r$, $0 < T < \frac{n}{r}$.

V. Discussion and Conclusion

This paper introduces more questions than it answers. We presented several distributed storage allocation problems and showed that despite the initial simplicity of the setup, there can be significant complexity. For the probability-1 recovery requirement, we established that the intuitive idea of spreading the budget maximally among all storage nodes is indeed optimal.

For the case of symmetric probabilistic allocations, we showed that it is optimal in the high-probability regime to maximally spread the given budget, by choosing $\ell = r$ which maximizes the probability of using each storage node while minimizing the amount of data stored in each nonempty node. As we vary the budget, we observe a sharp change in the optimal allocation — for small budgets and therefore low success probabilities, it is optimal to store the data object in its entirety (i.e. $\ell = 1$) and hope that the data collector accesses at least one of the nonempty storage nodes; for large budgets and therefore high success probabilities, it is optimal to store only $\frac{1}{r}$ amount of data in each used node and hope that the data collector accesses r of them. Our numerical investigation suggests that spreading minimally (i.e. $\ell = 1$) is optimal up to some budget T_{crit} and then spreading maximally (i.e. $\ell = r$) immediately becomes optimal without going through the intermediate choices of ℓ. This contrasts with deterministic allocations which, as discussed in the introduction, can have optimal symmetric allocations that involve an intermediate spreading of the budget.

It would be interesting to investigate the conditions under which nonsymmetric allocations are suboptimal — we conjecture that this is the case in the high-probability regime, which is also the regime of practical interest.

Another set of interesting problems involves the application of richer access models; for example, we can introduce a topology on the network of storage nodes and assume that the data collector accesses r nodes that are close to it.

Acknowledgments

We would like to thank Brighten Godfrey and Robert Kleinberg for introducing the problem to us and for sharing their insights on the optimality of nonsymmetric allocations. We also thank Dimitris Achlioptas for the interesting discussions.

References

[1] A. G. Dimakis, P. B. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran, "Network coding for distributed storage systems," *submitted for publication, preliminary version appeared in Proc. INFOCOM 2007.*

[2] A. Jiang, "Network coding for joint storage and transmission with minimum cost," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul. 2006.

[3] C. Fragouli, J.-Y. Le Boudec, and J. Widmer, "Network coding: An instant primer," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 1, pp. 63–68, Jan. 2006.

[4] T. Ho, M. Médard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.

[5] S. A. Aly, Z. Kong, and E. Soljanin, "Fountain codes based distributed storage algorithms for large-scale wireless sensor networks," in *Proc. ACM/IEEE Int. Conf. Inf. Process. Sensor Netw. (IPSN)*, Apr. 2008.

[6] A. G. Dimakis, V. Prabhakaran, and K. Ramchandran, "Ubiquitous access to distributed data in large-scale sensor networks through decentralized erasure codes," *Proc. Int. Symp. Inf. Process. Sensor Netw. (IPSN)*, Apr. 2005.

[7] A. Kamra, V. Misra, J. Feldman, and D. Rubenstein, "Growth codes: Maximizing sensor network data persistence," in *Proc. ACM SIGCOMM*, Sep. 2006.

[8] Y. Lin, B. Liang, and B. Li, "Data persistence in large-scale sensor networks with decentralized fountain codes," in *Proc. INFOCOM*, May 2007.

[9] R. Karp, R. Kleinberg, C. Papadimitriou, and E. Friedman, *Personal communication.*

[10] R. Kaas and J. M. Buhrman, "Mean, median and mode in binomial distributions," *Statistica Neerlandica*, vol. 34, no. 1, pp. 13–18, 1980.

[11] P. R. Beesack, "Improvements of Stirling's formula by elementary methods," *Univ. Beograd. Publ. Elektrotehn. Fak. Ser. Mat. Fiz.*, no. 274-301, pp. 17-21, 1969.