Graph based codes for distributed storage systems

July 21, 2015

Christine Kelley

University of Nebraska-Lincoln

Joint work with Allison Beemer and Carolyn Mayer

Combinatorics and Computer Algebra,
COCOA 2015, Fort Collins, CO.

Outline

1. Coding for distributed storage systems (DSS)

Outline

Outline

1. Coding for distributed storage systems (DSS)
2. Graph based codes
3. (r, ℓ)-locality of codes from expander graphs

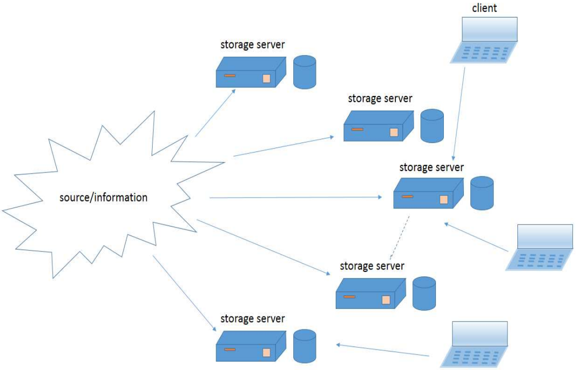
Outline

1. Coding for distributed storage systems (DSS)
2. Graph based codes
3. (r, ℓ)-locality of codes from expander graphs
4. Relation of locality to stopping sets

Outline

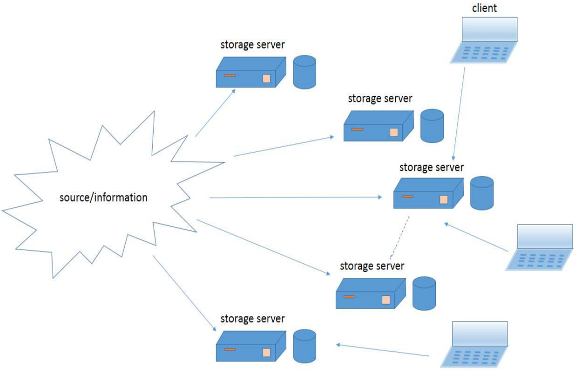
1. Coding for distributed storage systems (DSS)
2. Graph based codes
3. (r, ℓ)-locality of codes from expander graphs
4. Relation of locality to stopping sets
5. Ongoing work

Coding for distributed storage systems (DSS)



Goal: To store a lot of data across many servers so that multiple users can access the data reliably and efficiently

Coding for distributed storage systems (DSS)



Goal: To store a lot of data across many servers so that multiple users can access the data reliably and efficiently

- Amount of data increases faster than the hardware adapts.
- Errors are typically viewed as **erasures** (caused by server failures)

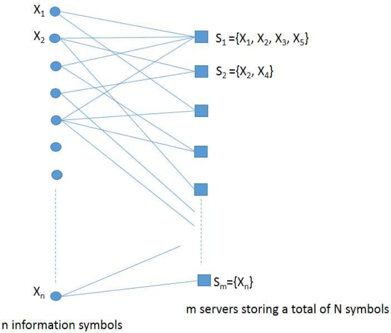
Some cost metrics

- Storage overhead
- Repair bandwidth
  - The number of bits communicated during repairs
- Locality
  - The number of nodes that participate in the repair process
- Availability
  - The number of parallel reads available for each data block

Since data centers are large, the idea is to do local repairs rather than full decoding (such as ML)

Replication based systems

Example: Combinatorial Batch Code



Rate is n/N where N is the total number of symbols stored across all servers.

Other code families

- Multiset batch codes:
  Stores linear combinations of data symbols and allows for multiple user access of same data request.
- Regenerating codes:
  Designed to reduce repair bandwidth.
- Fractional repetition codes:
  Allow for uncoded repairs of failed nodes while reducing repair bandwidth.
- Fractional repetition batch codes:
  Allow uncoded repairs and parallel reads of subsets of stored data.
- Locally repairable codes
  Systematic codes such that each information symbol has locality r.

Other code families

Many constructions use graphs and discrete structures.

In this talk, we will determine the (r, ℓ)-locality of two cases of codes based on expander graphs.

A code has (r, ℓ)-locality if any ℓ erased code symbols may be recovered by using at most r other intact code nodes.

Linear block codes

- An $[n, k, d]$ linear block code over $\mathbb{F}_2$ is a linear subspace of $\mathbb{F}_2^n$ with
  - codewords of length n
  - dimension k
  - rate $r = k/n$
  - minimum distance d

Linear block codes

- An $[n, k, d]$ linear block code over $\mathbb{F}_2$ is a linear subspace of $\mathbb{F}_2^n$ with
  - codewords of length n
  - dimension k
  - rate $r = k/n$
  - minimum distance d
- Generator matrix (Encoder): $G \in Mat_{k\times n}(\mathbb{F}_2)$

$$C = \{\mathbf{c} = \mathbf{x}G | \mathbf{x} \in \mathbb{F}_2^k\}.$$

- Parity-check matrix H is an $m \times n$ matrix such that $GH^T = \mathbf{0}$.

$$C = \{\mathbf{c} | \mathbf{c}H^T = \mathbf{0}\}.$$

Graph representation of a linear code

Let $\mathcal{C}$ be an $[n, k]$ linear block code defined by the following parity-check matrix H.

$H =$

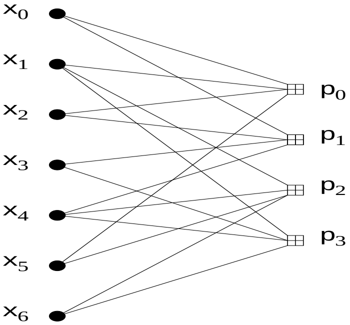
$parity \downarrow \backslash bit \rightarrow$	x_0	x_1	x_2	x_3	x_4	x_5	x_6
p_0	1	1	1	0	0	1	0
p_1	1	0	1	1	1	0	0
p_2	0	1	0	0	1	1	1
p_3	0	1	0	1	1	0	1

$$\mathbf{c}H^T = \mathbf{0} \Leftrightarrow \mathbf{c} \in \mathcal{C}$$

The code may be represented by the bipartite graph for which H is the incidence matrix.

Graph representation of a linear code

We can represent $\mathcal{C}$ by the following bipartite graph.

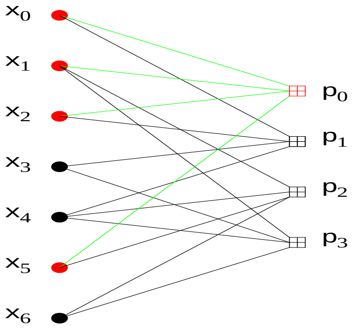


$$\mathbf{x}H^T = \mathbf{0} \Rightarrow$$
$$(x_0, x_1, \ldots, x_6)H^T = \mathbf{0}$$

Vertices on the left are called variable nodes, and vertices on the right are called constraint nodes.

Graph representation of a linear code

We can represent $\mathcal{C}$ by the following bipartite graph.



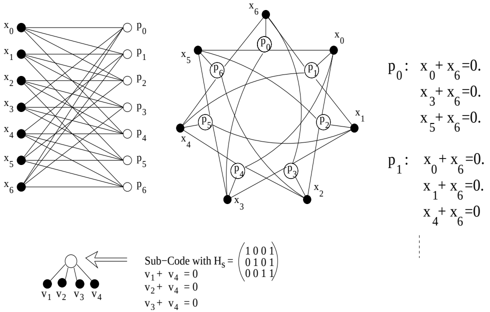
$$p_0 : \; x_0 + x_1 + x_2 + x_5 = 0.$$
$$p_1 : \; x_0 + x_2 + x_3 + x_4 = 0.$$
$$p_2 : \; x_1 + x_4 + x_5 + x_6 = 0.$$
$$p_3 : \; x_1 + x_3 + x_4 + x_6 = 0.$$

$\mathcal{C}$ is the set of all binary vectors that, when input to the variable nodes, satisfy all the check equations

Generalized graph based codes

Earlier, the constraint nodes represented simple parity checks.

A generalized graph based code with constraint nodes of degree d uses a "subcode" of block length d at each constraint (Tanner '81).



These have improved minimum distance and are useful in constructions arising from "nice" regular graphs.

Expander codes

- An expander code is a code whose graph is a good expander.
  - Good expanders have a large gap between the first and second largest eigenvalues of the associated adjacency matrix.
- Expander graphs have been used to design explicit asymptotically good codes (Sipser and Spielman '96).

Three standard cases of expander codes

1. Uses (c, d)-regular bipartite graph with m variable nodes and n constraint nodes where every subset U of variable nodes of size $< \alpha m$ has at least $\delta c|U|$ neighbors, for some $0 < \alpha < 1$ and $0 < \delta < 1$.
   - Either with simple parity constraints or subcode constraints.

2. Start with a d-regular graph G on n vertices with second largest eigenvalue μ. Let the edges represent the code symbols and the vertices represent subcode constraints.

3. Start with a (c, d)-regular bipartite graph with m left nodes and n right nodes and second largest eigenvalue μ. Let the edges represent code symbols and use two types of subcode constraints, one for each vertex set.
   - Special case when $c = d$

(Sipser and Spielman '96, Zemor '01, Janwa and Lal '02)

Three standard cases of expander codes

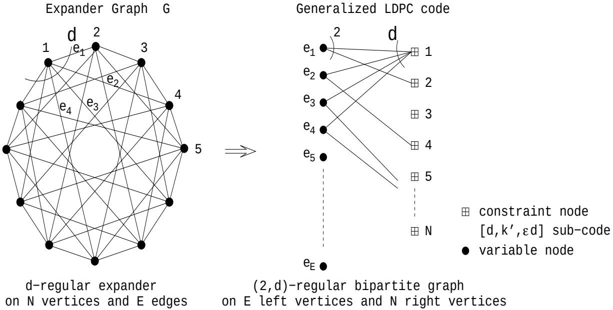
1. Uses (c, d)-regular bipartite graph with m variable nodes and n constraint nodes where every subset U of variable nodes of size $< \alpha m$ has at least $\delta c|U|$ neighbors, for some $0 < \alpha < 1$ and $0 < \delta < 1$.
   - Either with simple parity constraints or subcode constraints.

2. Start with a d-regular graph G on n vertices with second largest eigenvalue μ. Let the edges represent the code symbols and the vertices represent subcode constraints.

3. Start with a (c, d)-regular bipartite graph with m left nodes and n right nodes and second largest eigenvalue μ. Let the edges represent code symbols and use two types of subcode constraints, one for each vertex set.
   - Special case when $c = d$

(Rawat, Mazumdar, Vishwanath '14) derived (r, ℓ)-locality parameters for (1) and for case (3) when $c = d$.

Locality of d-regular graphs with subcode C



Theorem

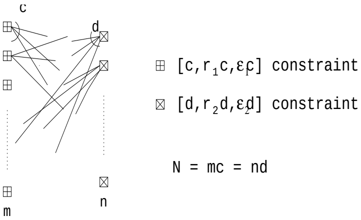
(Beemer, Mayer, K.) Let $\mathcal{C}(G, C)$ be the code defined by a d-regular graph G with second largest eigenvalue λ, and $[d, Rd, t+1]$ linear subcode C. Then $\mathcal{C}(G, C)$ has (r, ℓ)-locality for any

$$\ell \leq \frac{nd}{2}(\gamma^2 + \frac{\lambda}{d}(\gamma - \gamma^2)) \text{ where } \gamma \leq \frac{t-\lambda}{d-\lambda}, \text{ and } r = \ell d R.$$

(Proof similar to Rawat et al. paper)

Locality of (c, d)-regular graphs & subcodes C_1, C_2

Let $\mathcal{C} := \mathcal{C}(G, C_1, C_2)$ be the generalized code defined by a (c, d)-regular graph G on vertex sets M and N with $|M| = m$ left and $|N| = n$ right nodes and second largest eigenvalue μ. Let C_1 and C_2 be MDS subcodes with minimum distance d_1, d_2, respectively.



Theorem

(Beemer, Mayer, K.) The code $\mathcal{C}$ has (r, ℓ)-locality for

$$\ell \leq \frac{m\mu\epsilon}{2d} \min\{d_1, d_2\} \text{ and } r = \frac{\ell d}{2}(1 + rate(\mathcal{C}))$$

Locality of (c, d)-regular graphs & subcodes C_1, C_2

Algorithm
Input: A codeword with at most ℓ erasures.

1. Set $j = 0$, $i = 0$.
2. **while** not all erasures are corrected **do**
3. For every vertex $v \in M$ if $j \equiv 0 \mod 2$, or every vertex $v \in N$ if $j \equiv 1 \mod 2$, such that $1 \leq e \leq d_{j+1} - 1$ code symbols among c, or d, are in erasure, use erasure correcting algorithm for C_{j+1} to recover those erasures.
4. $i = i + 1$, $j \equiv i \mod 2$

Let S^i denote the set of vertices in M or N (depending on the iteration of the algorithm) that have at least one incident edge corresponding to an erased symbol at the end of iteration i.

Locality of (c,d)-regular graphs & subcodes C_1, C_2

Proof Sketch: Assume $d_i \geq (1+\epsilon)\mu$ for $i = 1, 2$ and $\ell \leq \frac{m\mu\epsilon}{2d} \min\{d_1, d_2\}$.

Claim: For $i \geq 1$,

$$|S^{i+1}| \leq \frac{|S^i|}{1+\epsilon}$$

Claim proof sketch: We prove relation for $i = 1$ and $i = 2$; later cases follow similar argument.

- Proof uses edge density variation (Janwa, Lal '02):

$$e(S,T) \leq \frac{d}{m}|S||T| + \frac{\mu}{2}(|S| + |T|)$$

- ℓ original erasures and $|S^1|$ vertices in N with at least d_1 adjacent erasures gives $d_1|S^1| \leq \ell$;
- Similarly, $|S^2|d_2 \leq e(S^1, S^2)$
- Applying assumptions and manipulating gives

$$|S^2| \leq \frac{d_1 m\mu}{d_1 m\mu(1+\epsilon)}|S^1| = \frac{|S^1|}{1+\epsilon}$$

Locality of (c, d)-regular graphs & subcodes C_1, C_2

- Similarly we get $|S^3| \leq \frac{|S^2|}{1+\epsilon}$

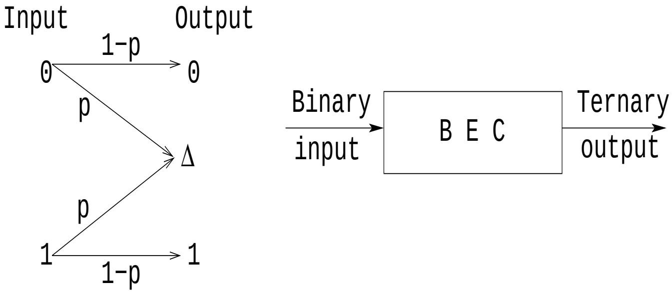
Since sequence $|S^1|, |S^2|, \ldots$ is decreasing, algorithm terminates.

Bound on r follows from a worst case argument:

- Correcting one erasure involves at most $\min\{cr_1, dr_2\}$ intact code symbols where r_i is rate of C_i.
- So $r \leq \ell \min\{cr_1, dr_2\}$
- It is easy to show that rate($\mathcal{C}$) $\geq r_1 + r_2 - 1$
- Assume WLOG $\max\{c, d\} = d$ and manipulate to obtain

$$\min\{cr_1, dr_2\} \leq \frac{d}{2}(1 + rate(\mathcal{C}))$$

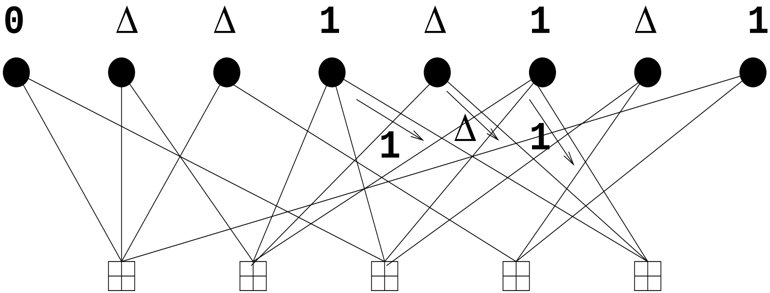
Example: the Binary Erasure Channel (BEC)



- Messages during algorithm are essentially bits.

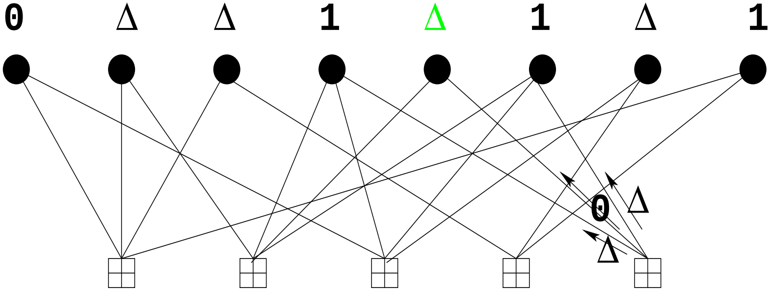
Example: the Binary Erasure Channel (BEC)

Received word $= (0, \Delta, \Delta, 1, \Delta, 1, \Delta, 1)$



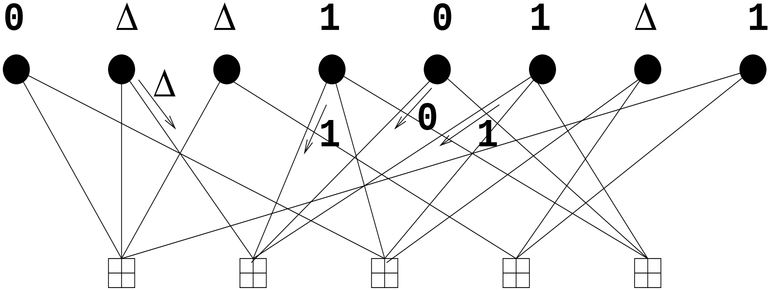
Example: the Binary Erasure Channel (BEC)

Received word $= (0, \Delta, \Delta, 1, \Delta, 1, \Delta, 1)$



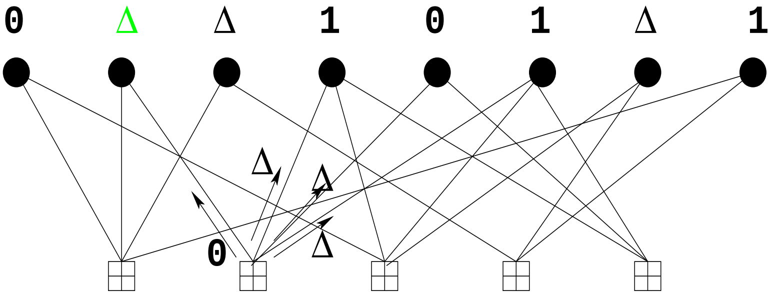
Example: the Binary Erasure Channel (BEC)

Received word = $(0, \Delta, \Delta, 1, \Delta, 1, \Delta, 1)$



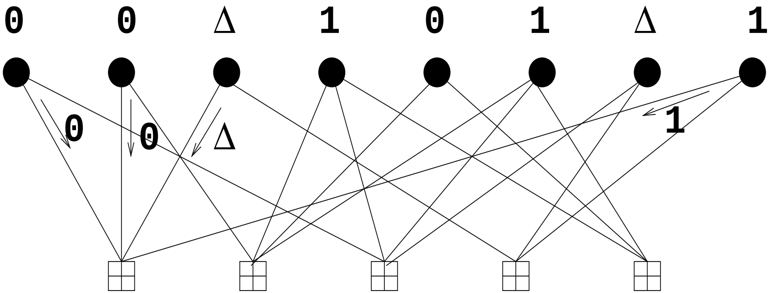
Example: the Binary Erasure Channel (BEC)

Received word $= (0, \Delta, \Delta, 1, \Delta, 1, \Delta, 1)$



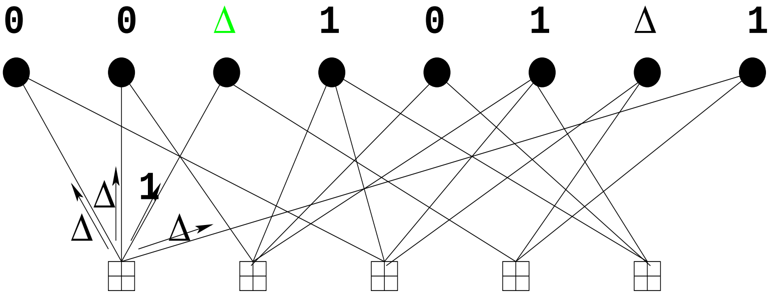
Example: the Binary Erasure Channel (BEC)

Received word $= (0, \Delta, \Delta, 1, \Delta, 1, \Delta, 1)$



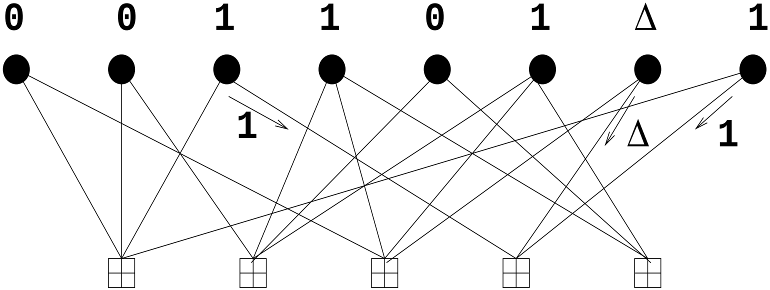
Example: the Binary Erasure Channel (BEC)

Received word $= (0, \Delta, \Delta, 1, \Delta, 1, \Delta, 1)$



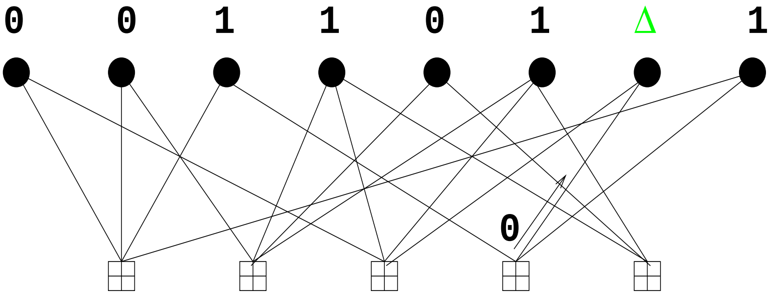
Example: the Binary Erasure Channel (BEC)

Received word = $(0, \Delta, \Delta, 1, \Delta, 1, \Delta, 1)$



Example: the Binary Erasure Channel (BEC)

Received word $= (0, \Delta, \Delta, 1, \Delta, 1, \Delta, 1)$



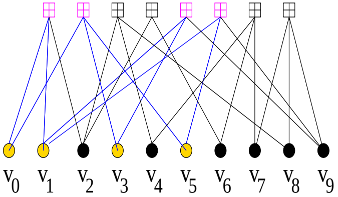
Example: the Binary Erasure Channel (BEC)

Received word $= (0, \Delta, \Delta, 1, \Delta, 1, \Delta, 1)$

$\longrightarrow$ **`Estimate=(0, 0, 1, 1, 0, 1, 0, 1)`**

Relation to stopping sets

- Iterative decoding failure of graph codes over erasure channel is characterized by stopping sets
- A stopping set S is a subset of the variable nodes such that each neighboring constraint of a vertex in S connects to S at least $d_{\min}$(subcode) times.



$S = \{v_0, v_1, v_3, v_5\}$

- When the code is decoded iteratively, $\ell \leq s_{\min} - 1$, where $s_{\min}$ is the size of the smallest stopping set.

Ongoing work

- Extending results to Hypergraph codes
- Determining the (r, ℓ)-locality of structured LDPC code families
  - Compare results to known bounds on $s_{\min}$.
- Examining the (r, ℓ)-locality of protograph LDPC codes.

Connections to other discrete structures

- Affine planes
- Transversal and resolvable designs
- Hypergraphs
- Turan graphs and cages
- Generalized polygons and other partial geometries
- Bipartite graphs with certain girth and constraints on vertex set sizes
- Certain families of regular and bi-regular graphs

Thank you!

- Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai. Batch codes and their applications. 2004
- M. Sipser and D. Spielman, Expander codes, 1996.
- G. Zemor, On expander codes, 2001.
- H. Janwa and A. K. Lal. On Tanner codes: minimum distance and decoding. 2002.
- A. Dimakis, A. Gal, A. Rawat, and Z. Song. Batch codes through dense graphs without short cycles. 2014.
- G. Kamath, N. Prakash, V. Lalitha, and P. V. Kumar. Codes with local regeneration. 2013.
- D. Papailiopoulos and A. Dimakis. Locally repairable codes. 2014.
- A. Rawat, A. Mazumdar, and S. Vishwanath. Cooperative local repair in distributed storage. 2014