Codes for Distributed Storage

- Two Recent Results

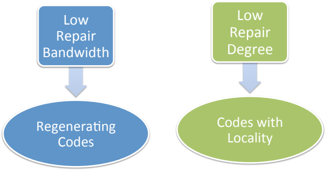
P. Vijay Kumar

joint work with
Birenjith Sasidharan and Gaurav Agarwal

Coding: From Practice to Theory
Simons Institute Workshop

February 11, 2015

Two Recent Results



- High-Rate MSR Code with Low Sub-Packetization Level (alphabet size)

  $$\alpha = (n-k)^{\frac{n}{(n-k)}}$$

- Codes with Hierarchical Locality

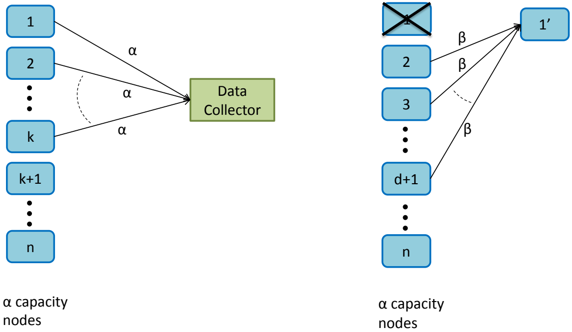
- A. G. Dimakis, P. B. Godfrey, Y. Wu, M. Wainwright, and K. Ramchandran, "Network Coding for Distributed Storage Systems," *IEEE Trans. Inform. Th.*, Sep. 2010.
- P. Gopalan, C. Huang, H. Simitci, and S. Yekhanin, "On the Locality of Codeword Symbols," *IEEE Trans. Inf. Theory*, Nov. 2012.

High-Rate MSR Codes

Birenjith Sasidharan, Gaurav Kumar Agarwal, and P. Vijay Kumar, "A High-Rate MSR Code With Polynomial Sub-Packetization Level," submitted to ISIT 2015, see also arXiv:1501.06662v1 [cs.IT] 27 Jan 2015.

Regenerating Codes

Parameters: $([n, k, d], \ [\alpha, \beta], \ B, \ \mathbb{F}_q)$

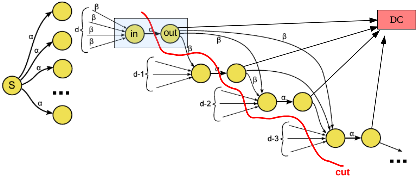


- Data Collection: Connect to any k nodes
- Nodes Repair: Connect to d nodes, download β symbols from each

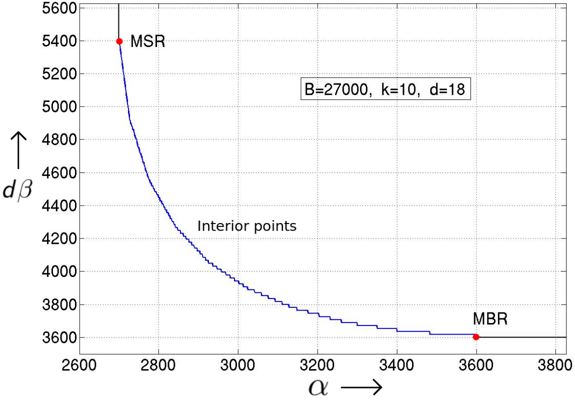
- We will assume Exact Node Repair

Cut-Set Bound from Network Coding

$$\text{File size } B \leq \sum_{i=1}^{k} \min\{\ \alpha,\ (d-i+1)\beta\ \}$$



Many Flavors of Optimality for Given (k, d, B)



- Repair Bandwidth ($d\beta$) vs Storage-per-Node (α)
- Two extremes:
  - Minimum Storage Regenerating (MSR) point
  - Minimum Bandwidth Regenerating (MBR) point

Constructions of MSR Codes (Rate $R \leq \frac{1}{2}$)

1. K. V. Rashmi, Nihar B. Shah and P. Vijay Kumar, "Optimal Exact-Regenerating Codes for Distributed Storage at the MSR and MBR Points via a Product-Matrix Construction," IT-Trans, August 2011.
2. Changho Suh and Kannan Ramchandran, "Exact-Repair MDS Code Construction Using Interference Alignment," IT-Trans, March 2011.
   - Nihar Shah, K. V. Rashmi, P. Vijay Kumar and Kannan Ramchandran, "Interference Alignment in Regenerating Codes for Distributed Storage: Necessity and Code Constructions," IT-Trans, April 2012.

Constructions of High-Rate MSR Codes (Rate $R > \frac{1}{2}$)

1. Viveck R. Cadambe, SyedAli Jafar, Hamed Maleki, Kannan Ramchandran and Changho Suh, "Asymptotic Interference Alignment for Optimal Repair of MDS Codes in Distributed Storage," IT-Trans, May 2013. (establish existence)
2. D. S. Papailiopoulos, A. G. Dimakis, and V. R. Cadambe, "Repair Optimal Erasure Codes through Hadamard Designs," IT-Trans, May 2013. (construction for 2 parities)
3. Itzhak Tamo, Zhiying Wang, and Jehoshua Bruck, "Zigzag Codes: MDS Array Codes With Optimal Rebuilding," IT-Trans, March 2013. (repair systematic nodes)
4. Z. Wang, I. Tamo, J. Bruck, "On Codes for Optimal Rebuilding Access," *Allerton*, 2011 (also repair parity)

Sub-Packetization Level

1. Bound[1] in [1]

$$\log_2(\alpha)\,(\log_\delta(\alpha)+1) \geq \frac{k-1}{2}$$
$$\delta = 1+\frac{1}{r-1}, \quad r=(n-k).$$

2. Construction in [2]

$$\alpha = r^{k+1}$$

3. Present Construction

$$\alpha = r^{\frac{n}{r}}$$

[1] Sreechakra Goparaju, Itzhak Tamo, and Robert Calderbank, "An Improved Sub-Packetization Bound for Minimum Storage Regenerating Codes," IT-Trans, May 2014.

[2] Z. Wang, I. Tamo, J. Bruck, "On Codes for Optimal Rebuilding Access," *Allerton*, 2011.

[1] Typo in presentation pointed out by E. Teletar has been corrected here.

Sub-Packetization Level

- Present Construction

$$\alpha = r^{\frac{n}{r}}$$
$$r = (n-k)$$

Parameter t	Rate $R = \frac{t-1}{t}$	Sub-packetization level α
$t = 3$	$\frac{2}{3}$	r^3
$t = 4$	$\frac{3}{4}$	r^4
$t = 5$	$\frac{4}{5}$	r^5

Construction Builds on the Earlier Work ...

- Itzhak Tamo, Zhiying Wang, and Jehoshua Bruck, "Zigzag Codes: MDS Array Codes With Optimal Rebuilding," IT-Trans, March 2013
- Z. Wang, I. Tamo, J. Bruck, "On Codes for Optimal Rebuilding Access," *Allerton*, 2011

How We WIll Explain Construction ...

- Parity-Check Point of View
- First present a simplistic view of parities that will repair but cannot handle data collection
- Will then refine this
- Will then refine this further (this will now permit data collection as desired)

Parameters of Construction

Parameters: ($[n, k, d]$, $[\alpha, \beta]$, B, $\mathbb{F}_q$)

n	tq
k	$(t-1)q$
d	$(n-1)$
α	q^t
β	q^{t-1}
r	q
Rate	$\frac{t-1}{t}$
α	$r^{\frac{n}{r}}$

Notation Used in Construction

Parameters: $(\; [n, k, d], \; [\alpha, \beta], \; B, \; \mathbb{F}_q \;)$

	Node 1	Node 2	$\cdots$	Node n
First symbol in node				
Second symbol in node				
$\vdots$	$\vdots$	$\vdots$	$\vdots$	
Last αth symbol in node				

$(n \times \alpha)$ codeword array

$$\text{Code symbol } C(\underbrace{\ell, \theta}_{\text{node}}; \underbrace{\underline{x}}_{\text{symbol in node}})$$

ℓth node group	θth node	$\underline{x}$th symbol
$\ell = 1, 2, \cdots, t$	$\theta \in \mathbb{F}_q$	$\underline{x} \in \mathbb{F}_q^t$

Parity Checks

Row-Sum Parity Checks:

$$\sum_{\ell=1}^{t} \sum_{\theta \in \mathbb{F}_q} C(\ell, \theta; \underline{z}) \quad = \quad 0$$

Jump (Zig-Zag) Parity Checks:

$$\sum_{\ell=1}^{t} \left(\sum_{\theta \neq z_\ell} C(\ell, \theta; \underline{z}) + C(\ell, z_\ell; \quad \underbrace{(\underline{z} - \Delta \underline{e}_\ell)}_{\text{jump in } \ell\text{th position}} \quad) \right) \quad = \quad 0$$

Illustrating Row-Sum Parity Checks ($z_1 = 0$ only)

$(x_1 x_2 x_3)$	$\ell = 1$		$\ell = 2$		$\ell = 3$	
	$\theta = 0$ Node 1	$\theta = 1$ Node 2	$\theta = 0$ Node 3	$\theta = 1$ Node 4	$\theta = 0$ Node 5	$\theta = 1$ Node 6
(000)	A	A	A	A	A	A
(001)	B	B	B	B	B	B
(010)	C	C	C	C	C	C
(011)	D	D	D	D	D	D
(100)						
(101)						
(110)						
(111)						

(A, B, C and D represent Row-Sum parity checks)

Illustrating Jump Parity Checks ($z_1 = 0$ only)

$(x_1x_2x_3)$	$\ell = 1$		$\ell = 2$		$\ell = 3$	
	$\theta = 0$ Node 1	$\theta = 1$ Node 2	$\theta = 0$ Node 3	$\theta = 1$ Node 4	$\theta = 0$ Node 5	$\theta = 1$ Node 6
(000)		P		P R		P Q
(001)		Q		Q S	P Q	
(010)		R	P R			R S
(011)		S	Q S		R S	
(100)	P					
(101)	Q					
(110)	R					
(111)	S					

- (P, Q, R and S represent Jump parity checks)
- From this it is clear how node 1 can be repaired by downloading 4 symbols from each of the other nodes

First refinement: Bringing in Coefficients

$$\sum_{\ell=1}^{t} \sum_{\theta \in \mathbb{F}_q} \underbrace{\lambda(\ell, \theta)}_{\text{coefficient}} C(\ell, \theta; \underline{z}) = 0$$

$$\sum_{\ell=1}^{t} \left(\sum_{\theta \neq z_\ell} \lambda(\ell, \theta) C(\ell, \theta; \underline{z}) + \lambda(\ell, z_\ell) C(\ell, z_\ell; \underbrace{(\underline{z} - \Delta \underline{e}_\ell)}_{\text{jump in } \ell\text{th position}}) \right) = 0$$

Second Refinement: Adding Extra Terms in the Parity Check Equations (for Data Collection)

$$\sum_{\ell=1}^{t} \sum_{\theta \in \mathbb{F}_q} \lambda(\ell,\theta) C(\ell,\theta;\underline{z}) = 0$$

$$\sum_{\ell=1}^{t} \left(\sum_{\theta \neq z_\ell} \lambda(\ell,\theta) C(\ell,\theta;\underline{z}) + \lambda(\ell,z_\ell) C(\ell,z_\ell; \underbrace{(\underline{z} - \Delta \underline{e}_\ell)}_{\text{jump in } \ell\text{th position}}) \right) + \underbrace{\sum_{\ell=1}^{t} \sum_{\theta \in \mathbb{F}_q} \gamma(\ell,\theta) C(\ell,\theta;\underline{z})}_{\text{helps guarantee data-collection property}} = 0$$

Parity-Check Matrix (without extra terms)

Associated parity-check matrix H is of the form:

	$\ell = 1$				$\ell = 2$				$\ell = 3$			
	$\theta = 0$		$\theta = 1$		$\theta = 0$		$\theta = 1$		$\theta = 0$		$\theta = 1$	
	Node 1		Node 2		Node 3		Node 4		Node 5		Node 6	
$z_1 = 0$	I_4		I_4		I_4		I_4		I_4		I_4	
$z_1 = 1$		I_4		I_4		I_4		I_4		I_4		I_4
$z_1 = 0$		I_4	I_4		A_1		A_3		A_5		A_7	
$z_1 = 1$		I_4	I_4			A_2		A_4		A_6		A_8

- $\Delta = 0$ in the first two rows
- $\Delta = 1$ (indicating jump parity) in bottom two rows

Parity-Check Matrix (with extra terms in blue)

To ensure data recovery, replace H by the form:

$$H = H_0 + H_1$$

where H_0, H_1 are given respectively by:

	$\ell = 1$				$\ell = 2$				$\ell = 3$			
	$\theta = 0$		$\theta = 1$		$\theta = 0$		$\theta = 1$		$\theta = 0$		$\theta = 1$	
	Node 1		Node 2		Node 3		Node 4		Node 5		Node 6	
$z_1 = 0$	I_4		I_4		I_4		I_4		I_4		I_4	
$z_1 = 1$		I_4		I_4		I_4		I_4		I_4		I_4
$z_1 = 0$	I_4		I_4		I_4		I_4		I_4		I_4	
$z_1 = 1$		I_4		I_4		I_4		I_4		I_4		I_4

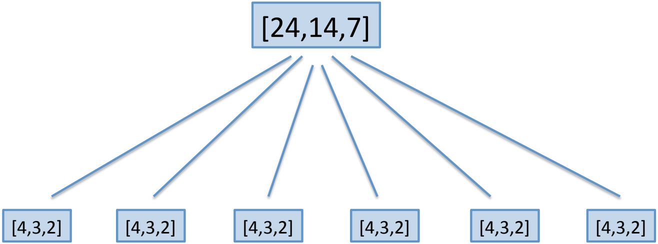
	$\ell = 1$				$\ell = 2$				$\ell = 3$			
	$\theta = 0$		$\theta = 1$		$\theta = 0$		$\theta = 1$		$\theta = 0$		$\theta = 1$	
	Node 1		Node 2		Node 3		Node 4		Node 5		Node 6	
$z_1 = 0$												
$z_1 = 1$												
$z_1 = 0$		I_4	I_4		A_1		A_3		A_5		A_7	
$z_1 = 1$		I_4	I_4			A_2		A_4		A_6		A_8

(this ensures the data collection property; Polynomial root counting)

Codes with Hierarchical Locality

Birenjith Sasidharan, Gaurav Kumar Agarwal, P. Vijay Kumar, “Codes With Hierarchical Locality,” submitted to ISIT 2015, see also arXiv:1501.06683 [cs.IT]

Codes with Locality do not Scale

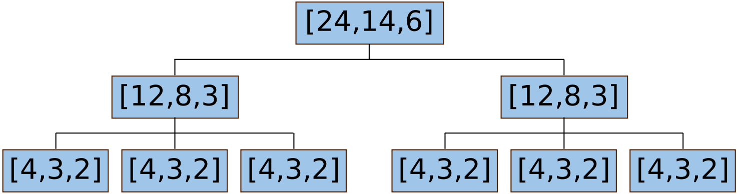


$$d \leq \underbrace{(n-k+1)}_{\text{Singleton bound}} - \underbrace{\left(\lceil \frac{k}{r} \rceil - 1\right)(\delta - 1)}_{\text{loss due to locality}}$$

r = locality

δ = minimum distance of the local code

Codes with Hierarchical Locality

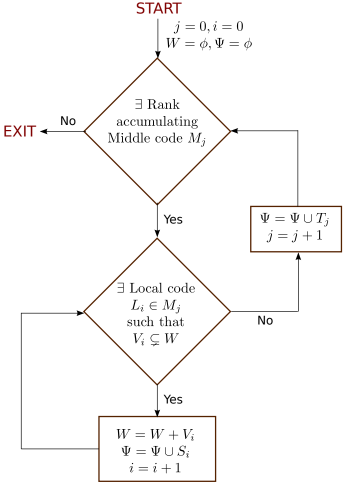


$$d \leq \underbrace{n - k + 1 - \left(\left\lceil \frac{k}{r_2} \right\rceil - 1 \right) (\delta_2 - 1)}_{\text{bound for codes with locality}} - \underbrace{\left(\left\lceil \frac{k}{r_1} \right\rceil - 1 \right) (\delta_1 - \delta_2)}_{\text{additional loss for 2nd locality layer}}$$

Bound on Minimum Distance

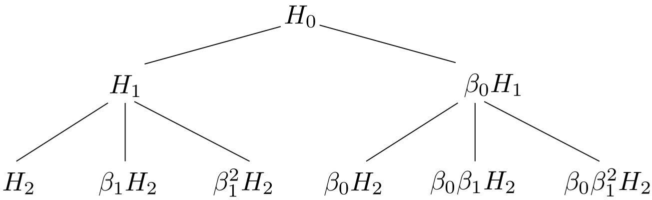
- Find a $(k-1)$-dimensional punctured code $\mathcal{C}_s$ with a large support.
- Then, $d_{\text{min}} \leq n - Supp(\mathcal{C}_s)$.

Algorithm used to identify $\mathcal{C}_s$



All-symbol Local Optimal Construction: An Example

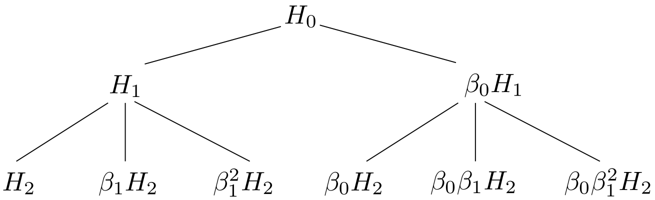
- Need to satisfy a divisibility condition $n_2 \mid n_1 \mid n$
- Example: $[24, 14]$, $[12, 8]$, $[4, 3]$.



1. Choose $\mathbb{F}_{25}$.
2. Identify subgroup chain $H_2 \subseteq H_1 \subseteq H$
3. Coset decomposition - supports of local codes

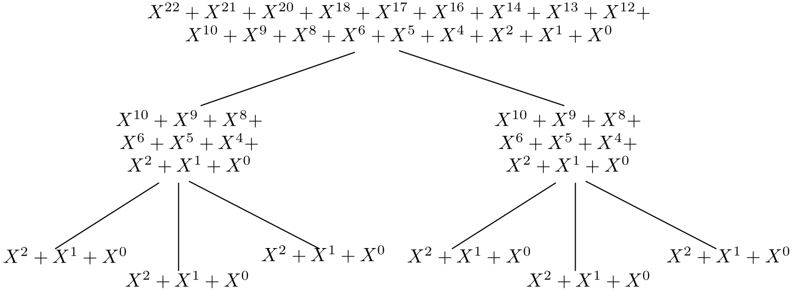
All-symbol Local Optimal Construction: An Example

- Need to satisfy a divisibility condition $n_2 \mid n_1 \mid n$
- Example: $[24, 14]$, $[12, 8]$, $[4, 3]$.

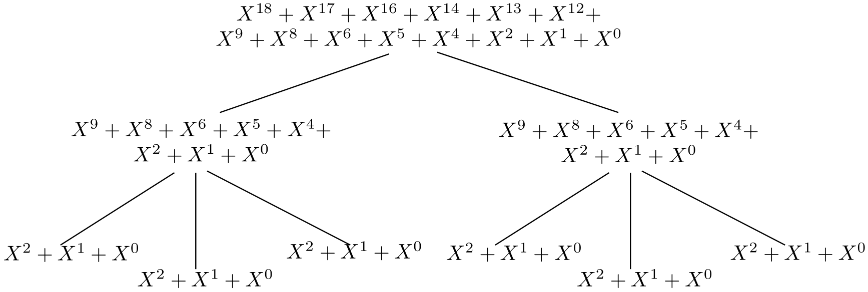


1. Choose $\mathbb{F}_{25}$.
2. Identify subgroup chain $H_2 \subseteq H_1 \subseteq H = \mathbb{F}_{25}^*$
3. Coset decomposition - supports of local codes

All-symbol Local Optimal Construction: An Example

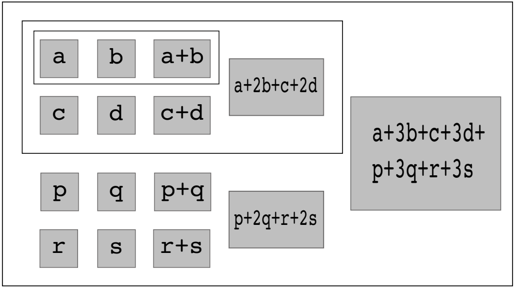


All-symbol Local Optimal Construction: An Example



Information-symbol Local Optimal Construction: Pyramid Codes

- $[n,k] = [15,8], \ [n_1, r_1] = [7,4], \ [n_2, r_2] = [3,2].$
- $\delta_2 = 2, \ \delta_1 = 3, \ d = 4.$ (optimal d_{min})



Pyramid Codes (contd.)

- Consider an MDS code with parameter $[k + d - 1, k, d] = [11, 8, 4]$.
- $G_{\text{mds}} = [I_{k\times k} \mid A_{k\times(d-1)}] = [I_{8\times 8} \mid A_{8\times 3}]$.

$$G^{s}_{\text{mds}} = \left[\begin{array}{c|c|c} I_{8\times 8} & \begin{array}{c} B_{4\times 2} \\ C_{4\times 2} \end{array} & D_{8\times 1} \end{array}\right],$$

$$G^{ss}_{\text{mds}} = \left[\begin{array}{c|c|c} I_{8\times 8} & \begin{array}{c|c} \begin{array}{c} E_{2\times 1} \\ F_{2\times 1} \end{array} & G_{4\times 1} \\ \hline \begin{array}{c} H_{2\times 1} \\ J_{2\times 1} \end{array} & K_{4\times 1} \end{array} & D_{8\times 1} \end{array}\right],$$

$$G_{\text{local}} = \left[\begin{array}{c|c|c} I_{8\times 8} & \begin{array}{cc|c|cc|c} E_{2\times 1} & & G_{4\times 1} & & & \\ & F_{2\times 1} & & & & \\ \hline & & & H_{2\times 1} & & K_{4\times 1} \\ & & & & J_{2\times 1} & \end{array} & D_{8\times 1} \end{array}\right]$$

Thanks!