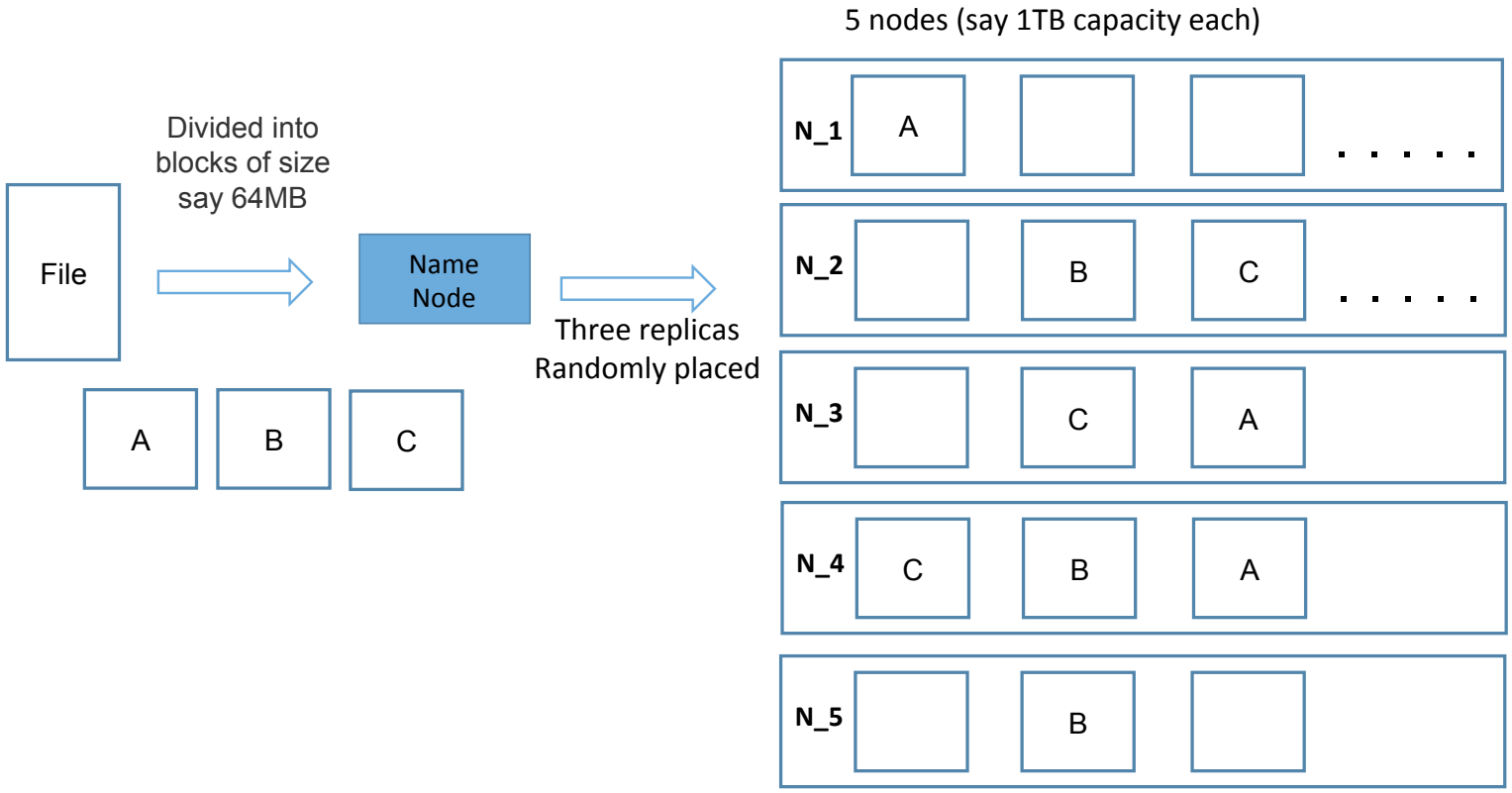
Erasure Coding for Distributed Storage Systems

Lalitha Vadlamani
Signal Processing and Communications Research Center,
IIIT Hyderabad

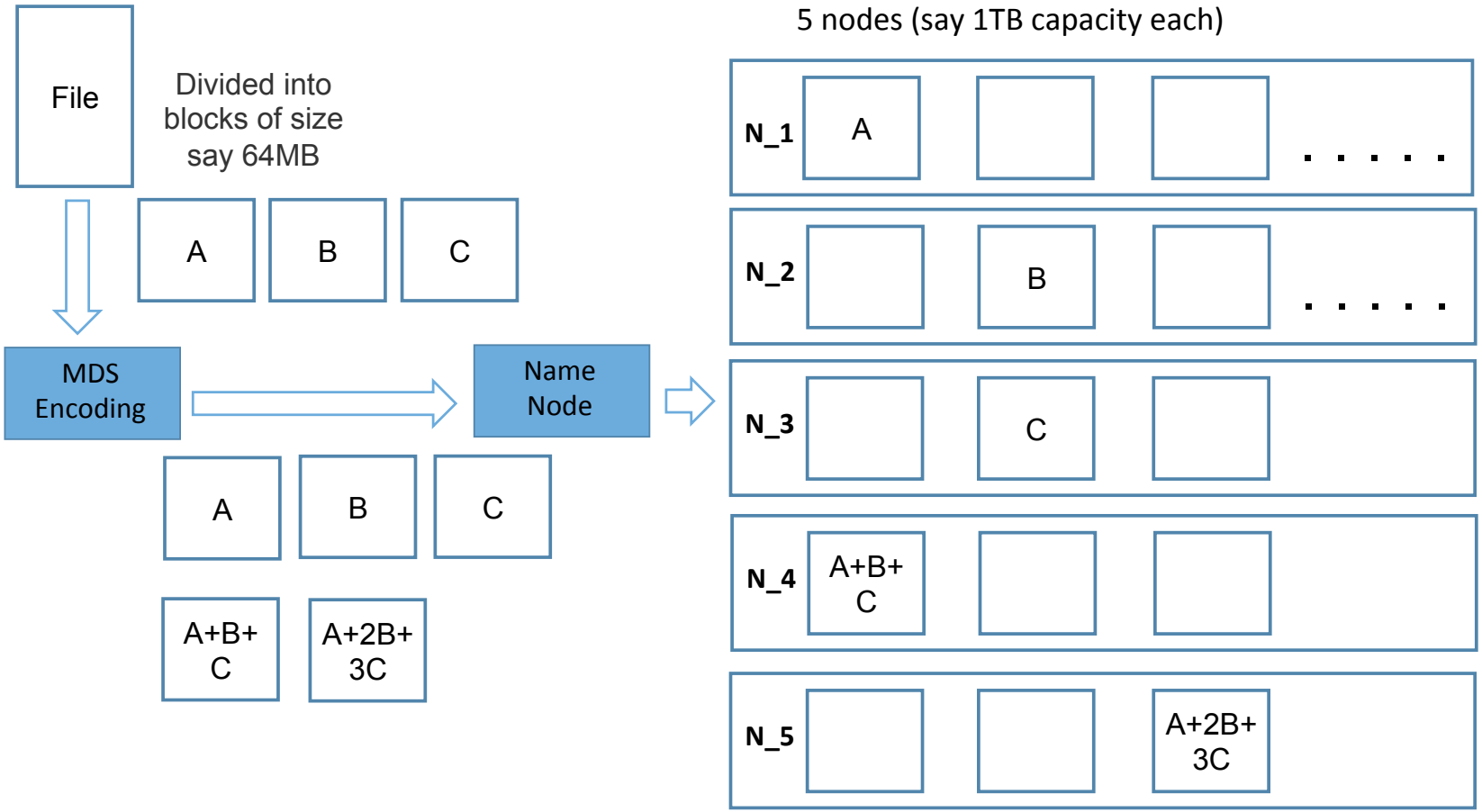
IndoSys
July 19, 2019

Distributed Storage System with Replication



- Typically hundreds of nodes
- The above process done for each file which enters the system

Distributed Storage System with Erasure Coding



- $(5, 3)$ Erasure code
- The blocks obtained after encoding placed in different nodes
- Encoding is done by dividing 64MB blocks into symbols of size 8 bits each

Types of Failures in a DSS

- Node is considered a failure domain
- Each encoded block is placed in a different failure domain (in this case different node)
- Permanent Failures: Data is lost because of hardware failure
- Temporary Failures: Power Outage, Software Upgrade. Data is temporarily unavailable but needs efficient recovery if there is a request for such data

Dealing with Failures

- Failures in distributed storage systems are modelled as erasures
- Erasure codes add redundancy to information so that even in the presence of failures, information is not lost
- We will deal with codes where redundant symbols are linear functions of the information symbols

Linear Block Codes

Definition (Generator Matrix)

Let $\mathbb{F}_q$ be a finite field with q elements. Let G be a $k \times n$ matrix with entries from $\mathbb{F}_q$. An n length codeword $\underline{c}$ corresponding to k length message $\underline{m}$ is given by $\underline{c} = \underline{m}G$. The set of all possible codewords obtained as $\underline{m}$ takes all q^k possible values is termed as linear block code $\mathcal{C}$.

- Block length of the code n
- Rate R of $\mathcal{C}$, $R = \frac{k}{n}$. Inverse of rate is storage overhead.
- Minimum distance

$$d_{\min}(\mathcal{C}) = \min\{d_H(\underline{x}, \underline{y}) | \underline{x}, \underline{y} \in \mathcal{C}, \underline{x} \neq \underline{y}\}$$

Dealing with Permanent Failures

- d_{min} determines the erasure tolerance capability of a code
- Precisely, a code with minimum distance d_{min} has no information lost, even in the presence of $d_{\text{min}} - 1$ erasures

Singleton Bound

- There is a tradeoff between storage overhead and erasure tolerance capability of a code

Theorem

For $[n, k, d_{\min}]$ *code,*

$$d_{\min} \leq n - k + 1.$$

A code whose $d_{\min}$ achieves the Singleton bound with equality is called a Maximum Distance Separable (MDS) code.

Reed Solomon Codes

- Let $\underline{m} = [m_0, \ldots m_{k-1}]$ be message vector over finite field $\mathbb{F}_q$
- Form the polynomial $f(x) = \sum_{i=0}^{k-1} m_i x^i$
- Pick $\alpha_i \in \mathbb{F}_q, 1 \leq i \leq n$ all distinct
- Codeword corresponding to $\underline{m}$ is $\underline{c} = [f(\alpha_1), \ldots, f(\alpha_n)]$
- This code can tolerate $n - k$ erasures ($k - 1$ degree polynomial can be uniquely determined by evaluations at k points)
- Minimum distance of RS code is $n - k + 1$

Vandermonde Matrices

Reed Solomon Code can also be described as

$$[c_1, \ldots, c_n] = [m_0, \ldots, m_{k-1}] \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_n^2 \\ \vdots & \vdots & & \vdots \\ \alpha_1^{k-1} & \alpha_2^{k-1} & \ldots & \alpha_n^{k-1} \end{bmatrix}$$

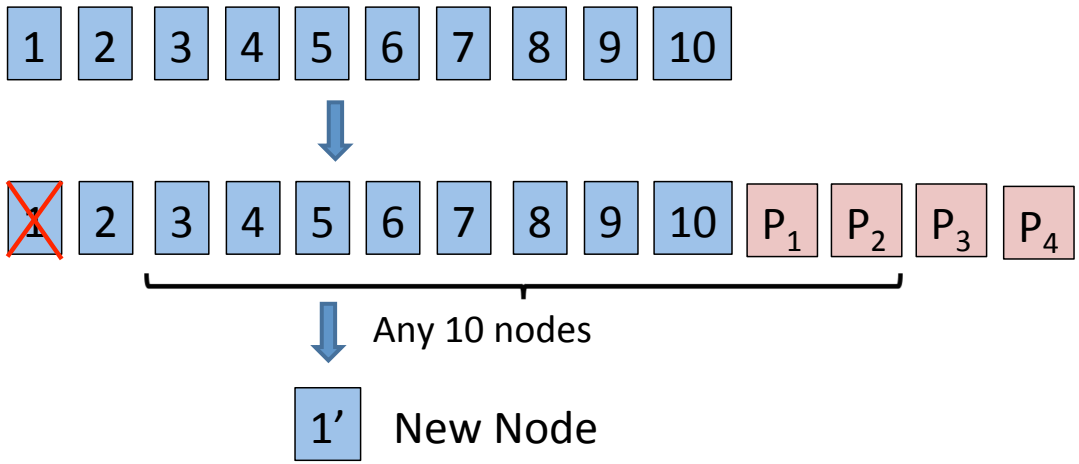
- Any k columns of Vandermonde matrix are linearly independent

Dealing with Temporary Failures

- 97% failures are single node failures (temporary failures)
- Replication is the best since another copy is available. However, storage overhead is high

Single Node Repair using MDS Codes

MDS codes are inefficient for single node repair



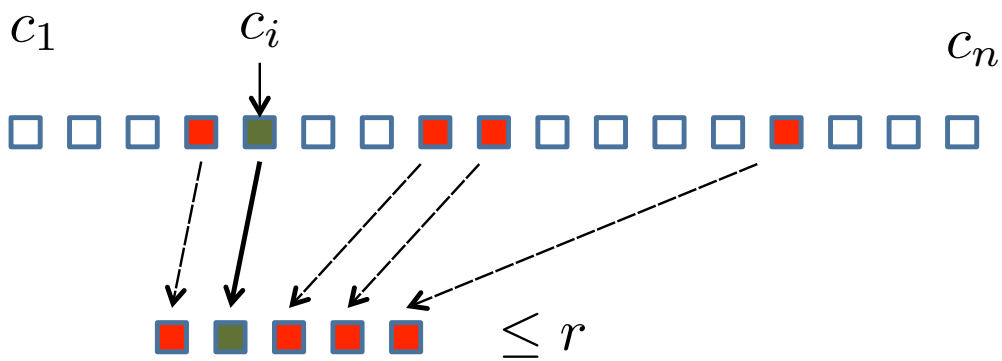
For node repair, the (naive) known strategy is:

- Connect to any 10 nodes
- Download 10 code symbols
- Reconstruct entire data file and then reconstruct data stored in the node

Locality Parameter

Another parameter known as locality parameter has been introduced to quantify the efficiency of dealing with single node failures

- Linear code $\mathcal{C}$ with parameters $[n, k, d_{\min}]$
- Code symbol c_i has locality r



- Consider a code in systematic form. The code is said to have **information locality** r if all the message symbols in the code have locality r

Singleton-like Bound

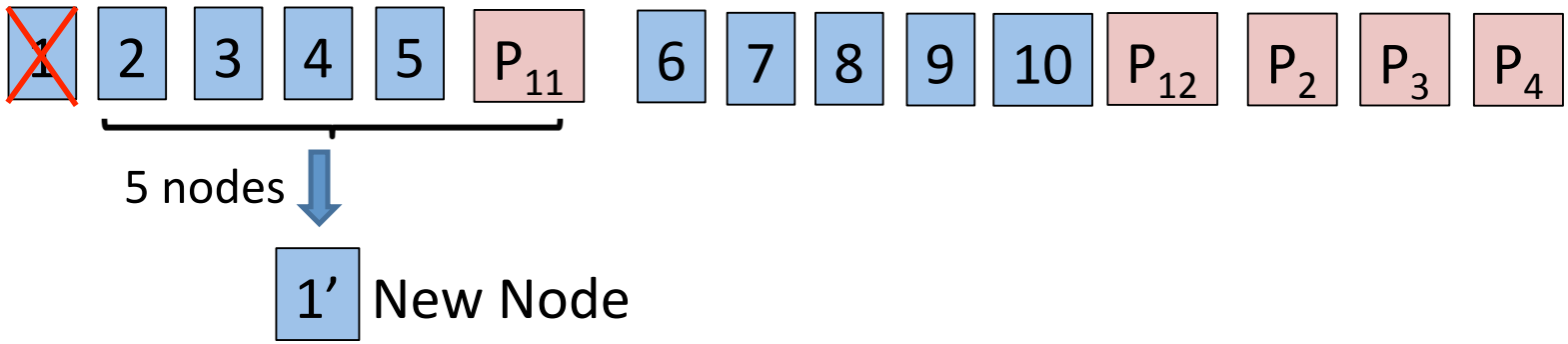
Theorem

For $[n, k, d_{\min}]$ *code with information locality* r

$$d_{min} \leq \underbrace{n - k + 1}_{\substack{\textit{Singleton} \\ \textit{bound}}} - \underbrace{\left(\left\lceil \frac{k}{r} \right\rceil - 1 \right)}_{\substack{\textit{Term due to} \\ \textit{locality constraint}}}$$

P. Gopalan, C. Huang, S. Yekhanin, H. Simitci, "On the Locality of Codeword Symbols," *IEEE Trans. Inform. Th.*, Nov. 2012. 2014 ComSoc/IT Joint Paper Award

Single Node Repair with LRC



- $P_1 = P_{11} + P_{12}$
- Repair by connecting to 5 nodes and hence efficient than MDS codes

Pyramid Code Construction via Example

- Given generator matrix G of a systematic $[11, 8, 4]$ MDS code:

$$G = \begin{bmatrix} 1 & & & & & g_{11} & g_{12} & g_{13} \\ & 1 & & & & g_{21} & g_{22} & g_{23} \\ & & \ddots & & & \vdots & \vdots & \vdots \\ & & & 1 & & g_{81} & g_{82} & g_{83} \end{bmatrix}$$

- Split first parity column, and then rearrange columns:

$$G' = \left[\begin{array}{cccc|cccc|cc} 1 & & & & g_{11} & & & & g_{12} & g_{13} \\ & 1 & & & g_{21} & & & & g_{22} & g_{23} \\ & & 1 & & g_{31} & & & & g_{32} & g_{33} \\ & & & 1 & g_{41} & & & & g_{42} & g_{43} \\ \hline & & & & & 1 & & & g_{52} & g_{53} \\ & & & & & & 1 & & g_{62} & g_{63} \\ & & & & & & & 1 & g_{72} & g_{73} \\ & & & & & & & & g_{82} & g_{83} \end{array}\right]$$

Optimality of Pyramid Code Construction

- The new $[12, 8, ?]$ code has two $[5, 4, 2]$ local codes.
- Minimum distance of code generated by G' is at least that generated by G. Thus $d_{\min} \geq 4$.
- Applying the bound on minimum distance,

$$\begin{aligned} d_{\min} &\leq n - k - \frac{k}{r} + 2 \\ &= 12 - 8 - \frac{8}{4} + 2 = 4 \end{aligned}$$

- Thus, $d_{\min} = 4$ and the pyramid code constructed is optimal

Pyramid Code Variants in Microsoft Products

LRC have been adopted throughout all Microsoft storage production lines, from the cloud to enterprise and the desktop. It was first deployed in Azure Storage in 2012.

In 2013, LRC also shipped with Windows Server 2012 R2 and Windows 8.1

Erasure Coding in HDFS

- Erasure codes on HDFS are implemented with striping - a data block composed of stripes.
- Encoding and decoding of a file can be configured by the erasure coding policy.
- Each policy is specified by
  - The EC schema: This includes the numbers of data and parity blocks in an erasure coding group (e.g., 6+3), as well as the actual code (e.g., Reed-Solomon, XOR).
  - The size of a stripe

`https://hadoop.apache.org/docs/r3.0.0/hadoop-project-dist/hadoop-hdfs/HDFSErasureCoding.html`

Erasure Coding in Ceph

- In Ceph, erasure codes have been implemented as plugins.
- Jerasure Erasure Code plugin contains an implementation of Reed-Solomon code based on the Jerasure and GF-Complete libraries.
- Locally Repairable Erasure Code plugin (LRC plugin) has been implemented in the reference below.

Kolosov, O., Yadgar, G., Liram, M., Tamo, I., and Barg, A., "On fault tolerance, locality, and optimality in locally repairable codes". In 2018 USENIX Annual Technical Conference (ATC 2018).

OpenEC

- A unified and configurable framework.
- Decouples erasure coding management from the storage workflows of distributed storage systems.
- Allows for specifying operations required in erasure coding through a directed-acyclic-graph-based programming abstraction.

Xiaolu Li, Runhui Li, Patrick P. C. Lee, and Yuchong Hu "OpenEC: Toward Unified and Configurable Erasure Coding Management in Distributed Storage Systems." Proceedings of the 17th USENIX Conference on File and Storage Technologies (FAST 2019).

Thanks!

Email: lalitha.v@iiit.ac.in