Expected Annual Fraction of Entity Loss as a Metric for Data Storage Durability

Ilias Iliadis
ili@zurich.ibm.com

April 24-28, 2023

Short Résumé

- Position
    - IBM Research - Zurich Laboratory since 1988

- Research interests
    - performance evaluation
    - optimization and control of computer communication networks
    - reliability of storage systems
    - storage provisioning for Big Data
    - cloud infrastructures
    - switch architectures
    - stochastic systems

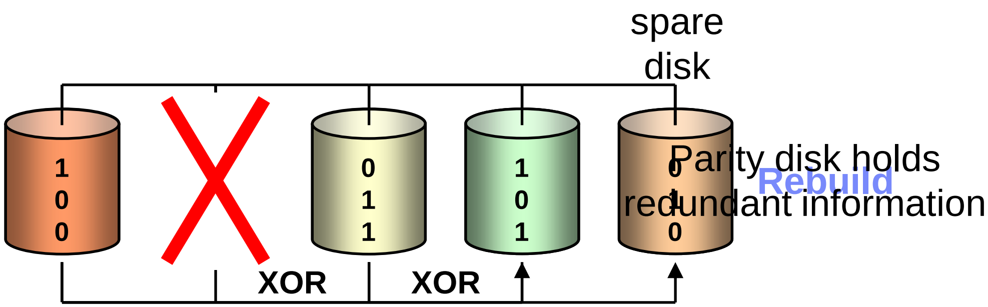
- Affiliations
    - IARIA Fellow
    - senior member of IEEE
    - IFIP Working Group 6.3

- Education
    - Ph.D. in Electrical Engineering from Columbia University, New York
    - M.S. in Electrical Engineering from Columbia University, New York
    - B.S. in Electrical Engineering from the National Technical University of Athens, Greece

Data Losses in Storage Systems

- Storage systems suffer from data losses due to
  - component failures
    - disk failures
    - node failures
  - media failures
    - unrecoverable and latent media errors

- Reliability enhanced by a large variety of redundancy and recovery schemes
  - RAID systems (**R**edundant **A**rray of **I**ndependent **D**isks)

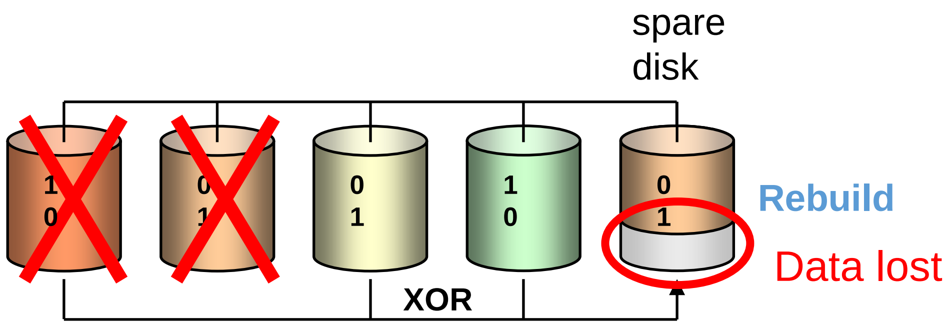


  - RAID-5: Tolerates one disk failure [Patterson *et al*. 1988]

Data Losses in Storage Systems

- Storage systems suffer from data losses due to
  - component failures
    - disk failures
    - node failures
  - media failures
    - unrecoverable and latent media errors

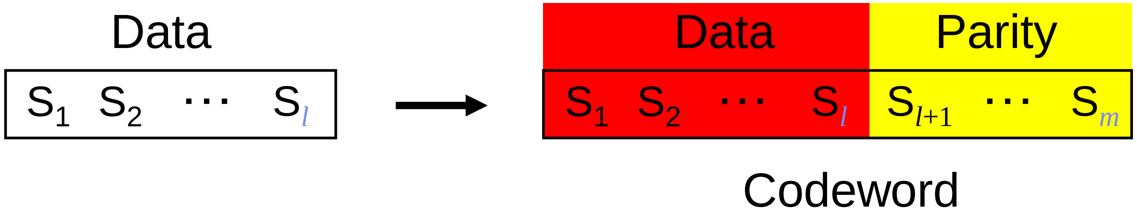
- Reliability enhanced by a large variety of redundancy and recovery schemes
  - RAID systems



  - RAID-5: Tolerates one disk failure
  - RAID-6: Tolerates two disk failures

Erasure Coded Schemes

- User data divided into blocks (symbols) of fixed size
  - Complemented with parity symbols
    - codewords



Data

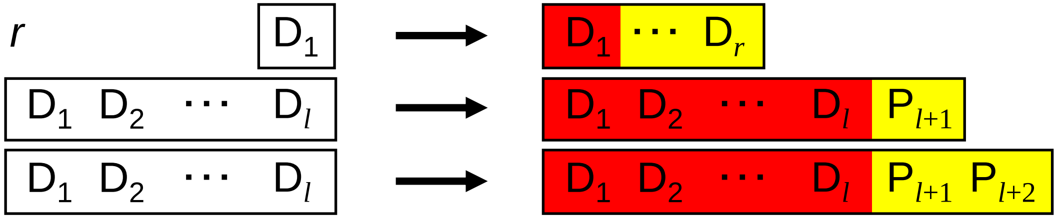
$S_1\ S_2 \cdots S_l$ →

Data | Parity

$S_1\ S_2 \cdots S_l$ | $S_{l+1} \cdots S_m$

Codeword

- (m,l) maximum distance separable (MDS) erasure codes
- Any subset of l symbols can be used to reconstruct a codeword
  - Replication : $l = 1$ and $m = r$
  - RAID-5 : $m = l + 1$
  - RAID-6 : $m = l + 2$



D_1 → $D_1 \cdots D_r$

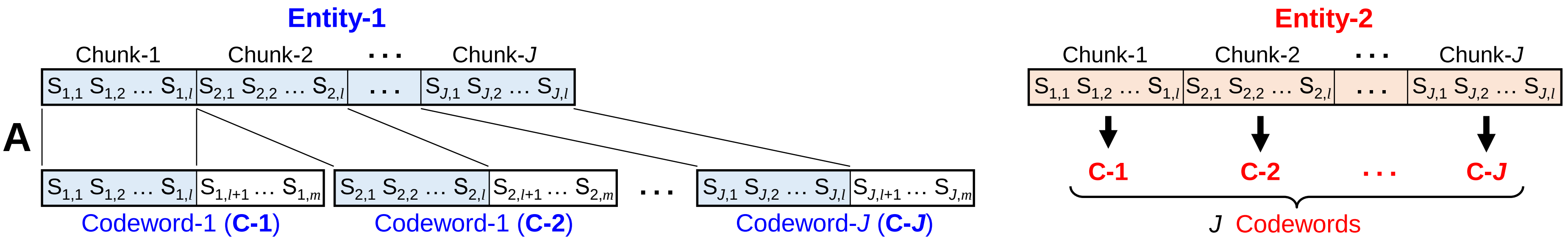
$D_1\ D_2 \cdots D_l$ → $D_1\ D_2 \cdots D_l$ P_{l+1}

$D_1\ D_2 \cdots D_l$ → $D_1\ D_2 \cdots D_l$ $P_{l+1}\ P_{l+2}$

- Storage efficiency : $s_{\text{eff}} = l/m$ (Code rate)

- Google : Three-way replication (3,1) → $s_{\text{eff}} = 33\%$ to Reed-Solomon (9,6) → $s_{\text{eff}} = 66\ \%$
- Facebook : Three-way replication (3,1) → $s_{\text{eff}} = 33\%$ to Reed-Solomon (14,10) → $s_{\text{eff}} = 71\ \%$
- Microsoft Azure : Three-way replication (3,1) → $s_{\text{eff}} = 33\%$ to LRC (16,12) → $s_{\text{eff}} = 75\ \%$

Formation of Codewords from Entity Data Contents



A

Entity-1

Chunk-1 Chunk-2 . . . Chunk-*J*

$S_{1,1}\ S_{1,2} \dots S_{1,l}$ | $S_{2,1}\ S_{2,2} \dots S_{2,l}$ | . . . | $S_{J,1}\ S_{J,2} \dots S_{J,l}$

$S_{1,1}\ S_{1,2} \dots S_{1,l}$ | $S_{1,l+1} \dots S_{1,m}$ — Codeword-1 (**C-1**)

$S_{2,1}\ S_{2,2} \dots S_{2,l}$ | $S_{2,l+1} \dots S_{2,m}$ — Codeword-1 (**C-2**)

. . .

$S_{J,1}\ S_{J,2} \dots S_{J,l}$ | $S_{J,l+1} \dots S_{J,m}$ — Codeword-*J* (**C-J**)

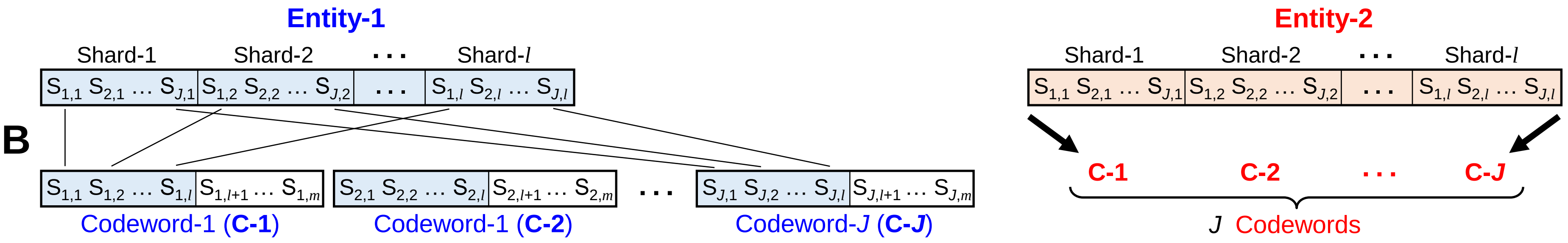
Entity-2

Chunk-1 Chunk-2 . . . Chunk-*J*

$S_{1,1}\ S_{1,2} \dots S_{1,l}$ | $S_{2,1}\ S_{2,2} \dots S_{2,l}$ | . . . | $S_{J,1}\ S_{J,2} \dots S_{J,l}$

C-1 **C-2** . . . **C-J**

J Codewords



B

Entity-1

Shard-1 Shard-2 . . . Shard-*l*

$S_{1,1}\ S_{2,1} \dots S_{J,1}$ | $S_{1,2}\ S_{2,2} \dots S_{J,2}$ | . . . | $S_{1,l}\ S_{2,l} \dots S_{J,l}$

$S_{1,1}\ S_{1,2} \dots S_{1,l}$ | $S_{1,l+1} \dots S_{1,m}$ — Codeword-1 (**C-1**)

$S_{2,1}\ S_{2,2} \dots S_{2,l}$ | $S_{2,l+1} \dots S_{2,m}$ — Codeword-1 (**C-2**)

. . .

$S_{J,1}\ S_{J,2} \dots S_{J,l}$ | $S_{J,l+1} \dots S_{J,m}$ — Codeword-*J* (**C-J**)

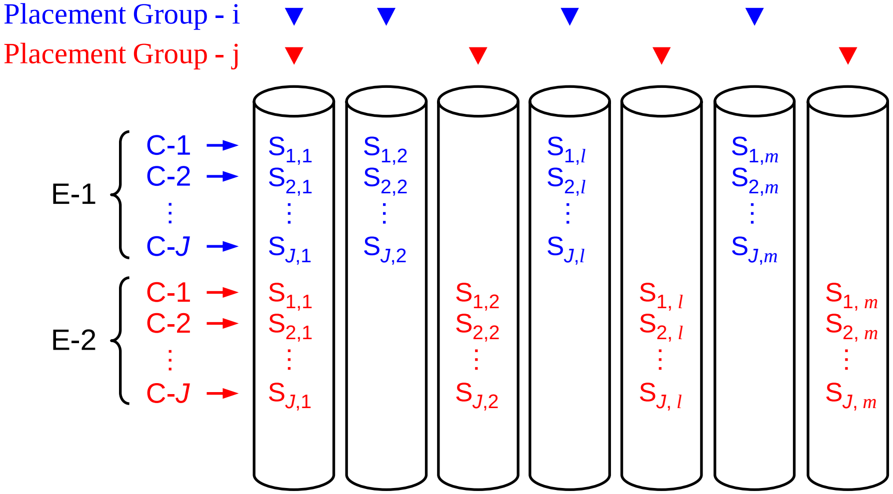
Entity-2

Shard-1 Shard-2 . . . Shard-*l*

$S_{1,1}\ S_{2,1} \dots S_{J,1}$ | $S_{1,2}\ S_{2,2} \dots S_{J,2}$ | . . . | $S_{1,l}\ S_{2,l} \dots S_{J,l}$

C-1 **C-2** . . . **C-J**

J Codewords



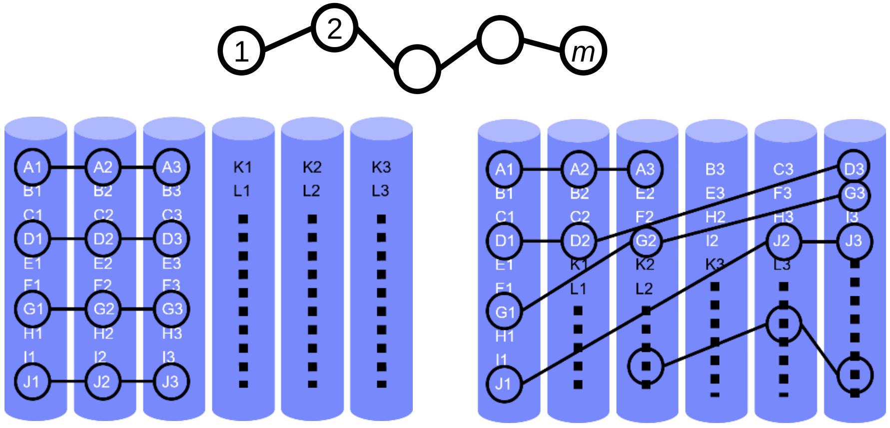
Placement Group - i

Placement Group - j

E-1: C-1, C-2, ⋮, C-J

E-2: C-1, C-2, ⋮, C-J

Entity stored in a Placement Group



Clustered placement

Declustered placement

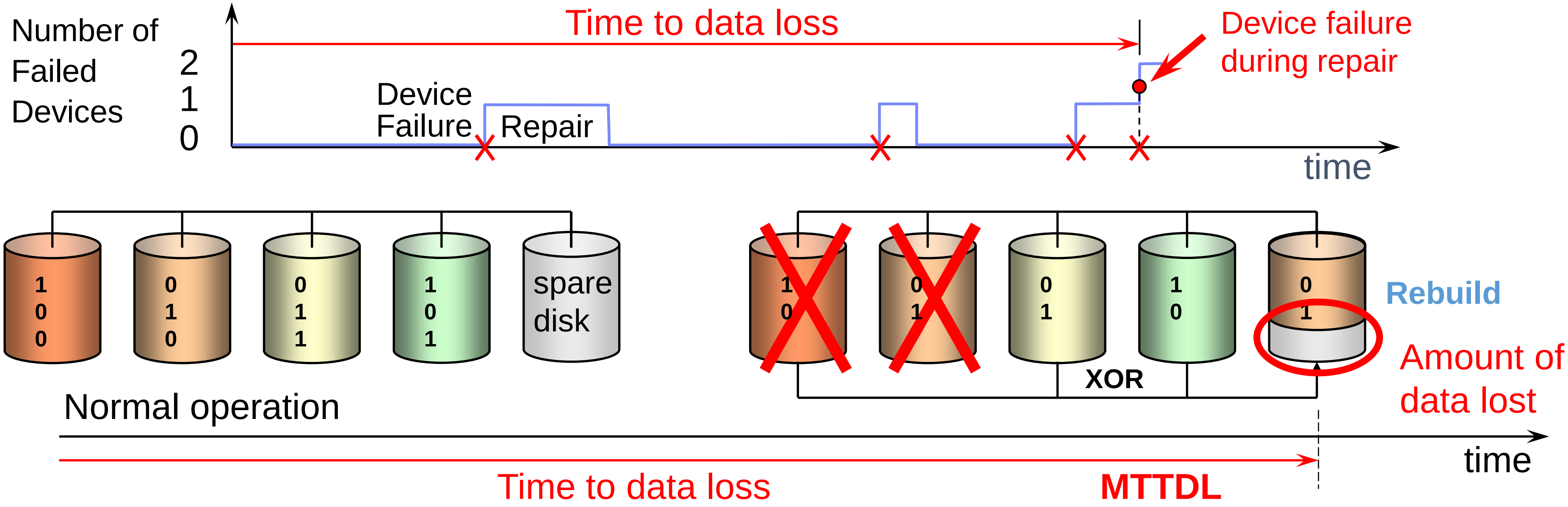
Codeword and Entity Loss

- Erasure coding
  - reduction in storage overhead
  - improvement of reliability achieved

  but
  - repair problem
    - increased network traffic needed to repair data lost
    - Solution: lazy rebuild
      - rebuild process not triggered immediately upon first device failure
      - rebuild process delayed until additional device failures occur
        - ✓ reduces recovery bandwidth
        - ✓ keeps the impact on read performance and data durability low

- **Fixed-size entities**
  - each entity contains J codewords
  - when a codeword of an entity loses $m - l + 1$ or more symbols, this codeword, and consequently the entity is permanently lost
    - Permanent codeword loss $\Rightarrow$ Permanent entity loss
  - reconstruction of successive codewords leads to the successive reconstruction of entities

Reliability Metrics – MTTDL and EAFDL



- Data loss events documented in practice by Yahoo!, LinkedIn, Facebook and Amazon
  - Amazon S3 (Simple Storage Service) is designed to provide 99.999999999% durability of objects over a given year
    - average annual expected loss of a fraction of 10^{-11} of the data stored in the system
- Assess the implications of system design choices on the
  - frequency of data loss events
    - **Mean Time to Data Loss (MTTDL)**
  - amount of data lost
    - **Expected Annual Fraction of Data Loss (EAFDL)**

      I. Iliadis and V. Venkatesan,
      "Expected Annual Fraction of Data Loss as a Metric for Data Storage Reliability", MASCOTS 2014
  - These two metrics provide a useful profile of the magnitude and frequency of data losses

Reliability of Erasure Coded Systems

- Analytical closed-form expressions for the MTTDL and EAFDL of erasure coded systems in the presence of latent errors when the lazy rebuild scheme is employed

  I. Iliadis, “Effect of Lazy Rebuild on Reliability of Erasure-Coded Storage Systems”, CTRQ 2022

  - General method for obtaining the MTTDL and EAFDL
    - Most likely path that leads to data loss
      - direct path to data loss

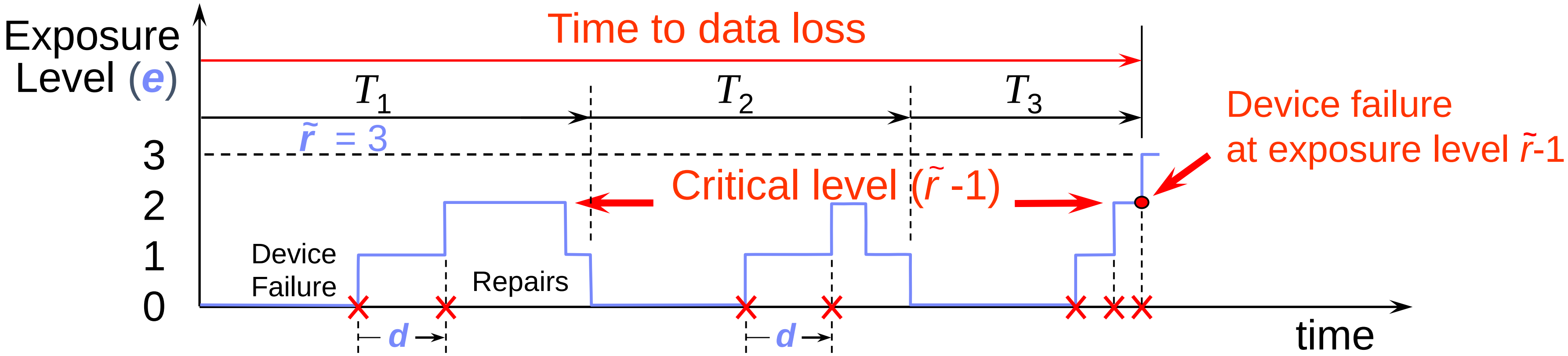
OBJECTIVE

To theoretically analyze the Expected Annual Fraction of Entity Loss (EAFEL)

RESULTS

- General method for obtaining the EAFEL
- Evaluation of EAFEL in parallel with MTTDL
  - Analytical approach that does not involve any Markov analysis
    - EAFEL and MTTDL tend to be insensitive to the failure time distributions
      - Real-world distributions, such as Weibull and gamma

Non-Markov Analysis for MTTDL and EAFEL



- EAFEL evaluated in parallel with MTTDL
  - $\tilde{r}$: Minimum number of device failures that may lead to data loss ($\tilde{r} = m - l + 1$)
  - d : Lazy rebuild threshold ($0 \le d < m - l$)
  - e : Exposure Level: maximum number of symbols that any codeword has lost
  - T_i : Cycles (Fully Operational Periods / Repair Periods)
  - P_{DL} : Probability of data loss during repair period
  - Y : Number of lost entities upon a first-device failure
  - J : Number of codewords per entity
  - N_E : Number of entities stored in a system comprised of n devices
  - $1/\lambda$: Mean Time to Failure (MTTF) of a device

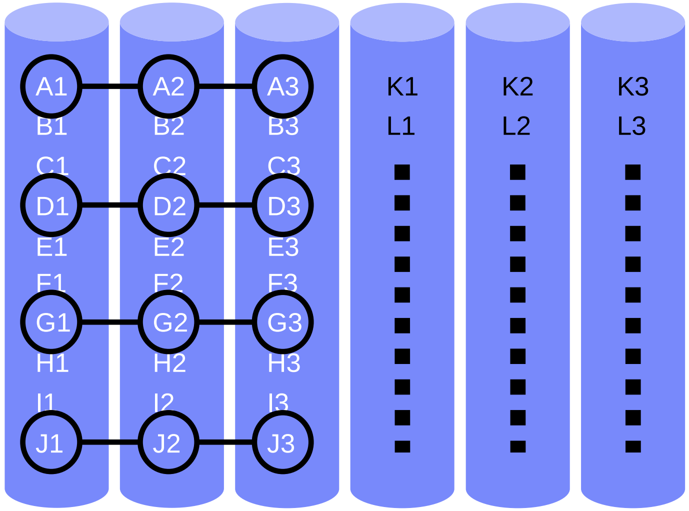
➢ $\text{MTTDL} = \sum_i E(T_i) = \dfrac{E(T)}{P_{\mathrm{DL}}}$ EAFEL $\approx \dfrac{E(Y)}{E(T)\, NE}$

- System evolution does not depend only on the latest state, but on the entire path
  - ➢ underlying models are not semi-Markov

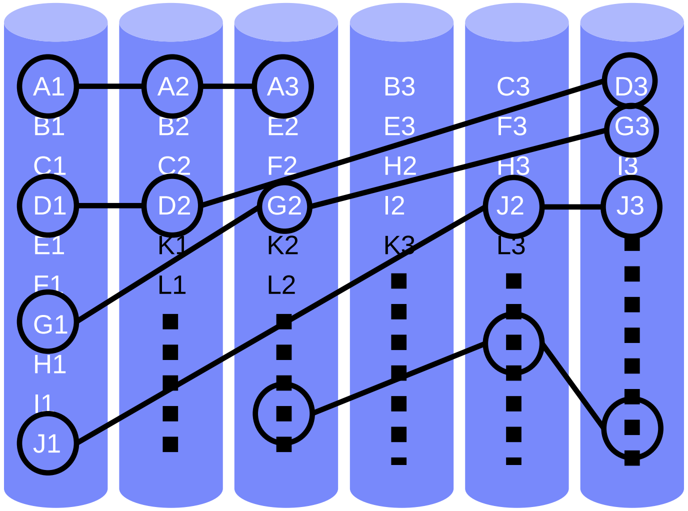
MTTDL and EAFEL expressions obtained using non-Markov analysis

Redundancy Placement

Erasure code with codeword length 3



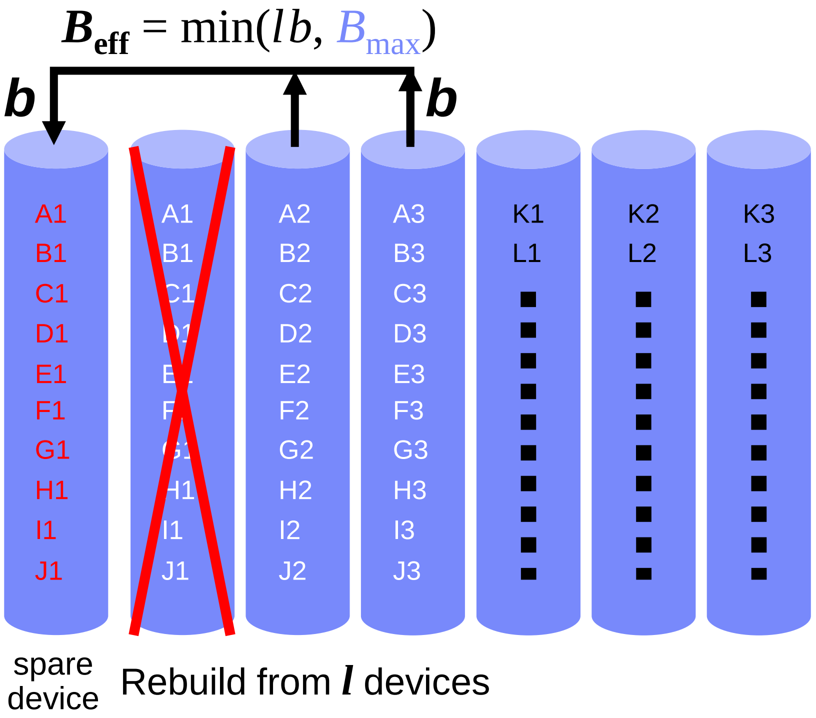
Clustered Placement



Declustered Placement

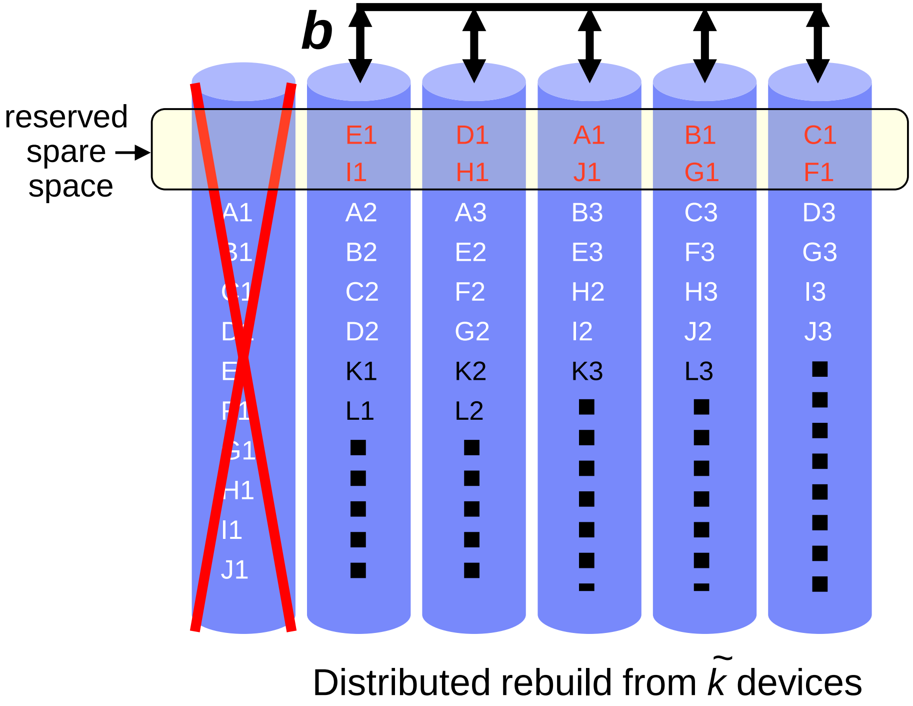
Device Failure and Rebuild Process

$$B_{\mathrm{eff}} = \min(lb, B_{\max})$$



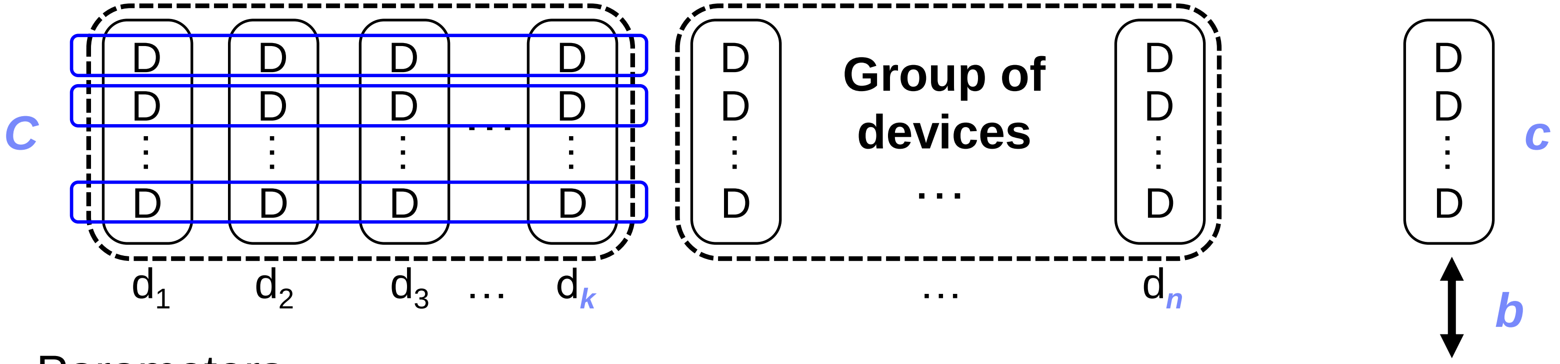
Clustered Placement

$$B_{\mathrm{eff}} = \min(\tilde{k}b, B_{\max})$$



Declustered Placement

System Model



- Parameters
  - n : number of storage devices
  - k : number of devices in a group
  - c : amount of data stored on each device
  - C : number of codeword symbols stored in a device
  - b : average reserved rebuild bandwidth per device

  - $1/\lambda$: Mean Time to Failure (MTTF) of a device
    - General non-exponential failure distributions
  - $1/\mu$: Time to read (or write) an amount of c data at a rate b from (or to) a device
    - $1/\mu = c / b$

➢ Highly reliable devices: $\lambda / \mu \ll 1$

Theoretical Results

- n : number of storage devices
- k : group size (number of devices in a group)
- c : amount of data stored on each device
- (m,l) : MDS erasure code
- d : lazy rebuild threshold
- b : reserved rebuild bandwidth per device
- $B_{\max}$: Maximum network rebuild bandwidth per group of devices
- $1/\lambda$: mean time to failure of a storage device
- P_s : probability of an unrecoverable sector (symbol) error

$$\mathrm{MTTDL} \approx \frac{E(T)}{P_{\mathrm{DL}}} \quad \text{and} \quad \mathrm{EAFEL} \approx \frac{E(Y)}{E(T) \cdot N_E} \quad \text{where}$$

$$P_{\mathrm{DL}} \approx P_{\mathrm{DF}} + \sum_{u=d+1}^{\tilde{r}-1} P_{\mathrm{UF}_u}$$

$$P_{\mathrm{UF}_u} \approx -\left(\lambda\, c \prod_{j=1}^{d} V_j\right)^{u-d-1} \frac{E(X^{u-d-1})}{[E(X)]^{u-d-1}} \left(\prod_{i=d+1}^{u-1} \frac{\tilde{n}_i}{b_i}\, V_i^{u-1-i}\right) \log(\hat{q}_u)^{-(u-d-1)} \left(\hat{q}_u - \sum_{i=0}^{u-d-1} \frac{\log(\hat{q}_u)^i}{i!}\right)$$

$$P_{\mathrm{DF}} \approx \frac{(\lambda\, c \prod_{j=1}^{d} V_j)^{\tilde{r}-d-1}}{(\tilde{r}-d-1)!} \frac{E(X^{\tilde{r}-d-1})}{[E(X)]^{\tilde{r}-d-1}} \prod_{i=d+1}^{\tilde{r}-1} \frac{\tilde{n}_i}{b_i}\, V_i^{\tilde{r}-1-i} \,, \qquad E(T) = \left(\sum_{u=0}^{d} \frac{1}{\tilde{n}_u}\right) \Big/ \lambda$$

$$N_E = \frac{n}{m} \cdot \frac{C}{J}$$

$$E(Y) \approx E(Y_{\mathrm{DF}}) + \sum_{u=d+1}^{\tilde{r}-1} E(Y_{\mathrm{UF}_u})$$

$$E(Y_{\mathrm{UF}_u}) \approx \frac{C}{J} \frac{P_u}{u-d} \left(\prod_{j=1}^{u-1} V_j\right) \tilde{q}_u \qquad \tilde{q}_u = 1 - q_u^{f_{\mathrm{cor}}\, J}$$

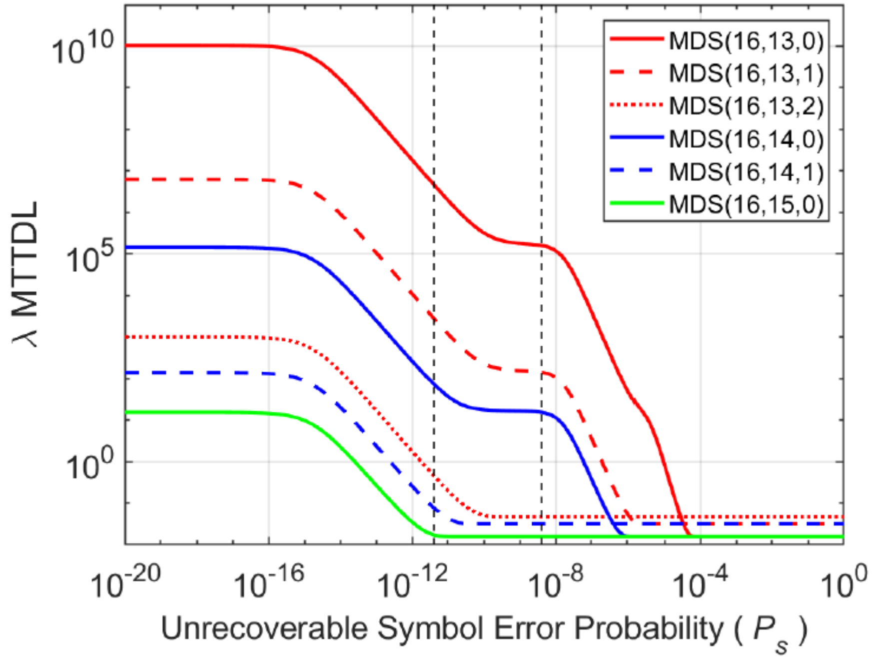
$$E(Y_{\mathrm{DF}}) \approx \frac{C}{J} \frac{P_{\mathrm{DF}}}{\tilde{r}-d} \prod_{j=1}^{\tilde{r}-1} V_j \qquad q_u = 1 - \sum_{j=\tilde{r}-u}^{m-u} \binom{m-u}{j} P_s^j (1-P_s)^{m-u-j}$$

Numerical Results

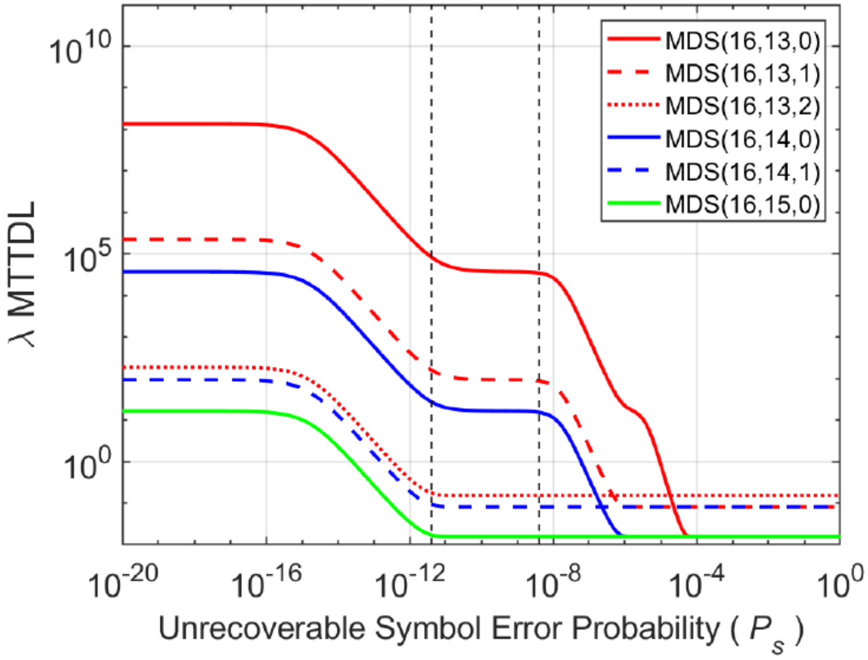
- n = 64 : number of storage devices
- c = 20 TB : amount of data stored on each device
- s = 512 B : sector size
- $1/\lambda$ = 876,000 h : MTTF
- b = 100 MB/s : reserved rebuild bandwidth
  - ➢ $1/\mu = c/b$ = 55.5 h : MTTR
  - ➢ $\lambda/\mu = 6\text{x}10^{-5} \ll 1$: MTTR to MTTF ratio
- m = 16 : number of symbols per codeword
- P_s : P (unrecoverable sector error)

- Numerical results for two system configurations
  - Declustered placement
    - ➢ $k = n = 64$
  - Clustered placement
    - ➢ $k = 16$
      - System comprises 4 clustered groups

Effect of Latent Errors on MTDDL



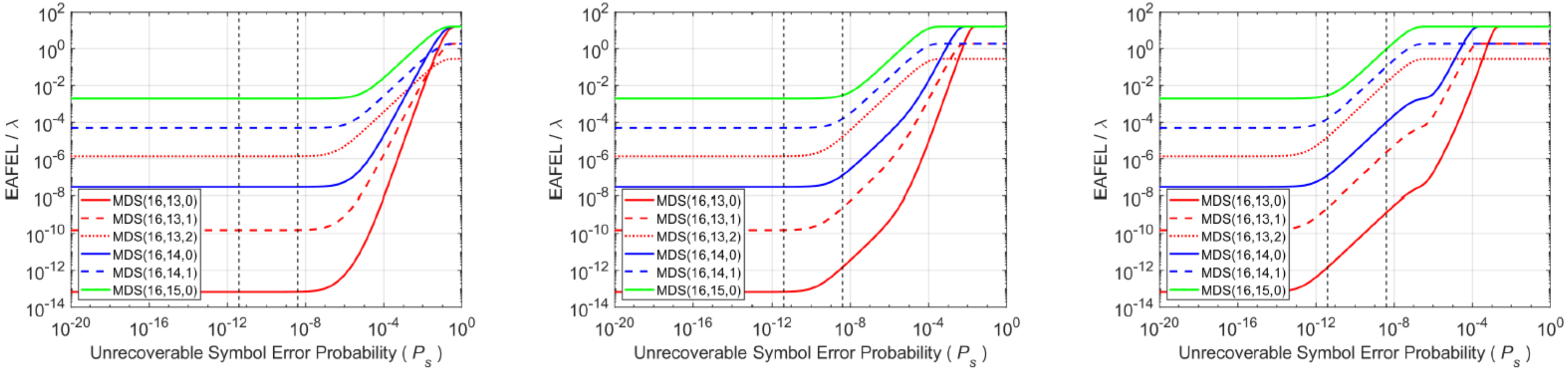
(a) $k = 64$ (declustered data placement scheme)



(b) $k = 16$ (clustered data placement scheme)

- MTTDL is not dependent on the entity size
- MTTDL decreases monotonically with P_s and exhibits $m - l - d$ plateaus
- Field measurements show P_s to be in the interval [4.096×10^{-12}, 4.096×10^{-9}]
- MTTDL significantly degraded by the presence of latent errors
- Increasing the number of parities (reducing l) improves reliability by orders of magnitude
- Employing lazy rebuild degrades reliability by orders of magnitude
- The declustered placement scheme achieves a significantly higher MTTDL than the clustered one

Effect of Latent Errors on EAFEL for Declustered Placement



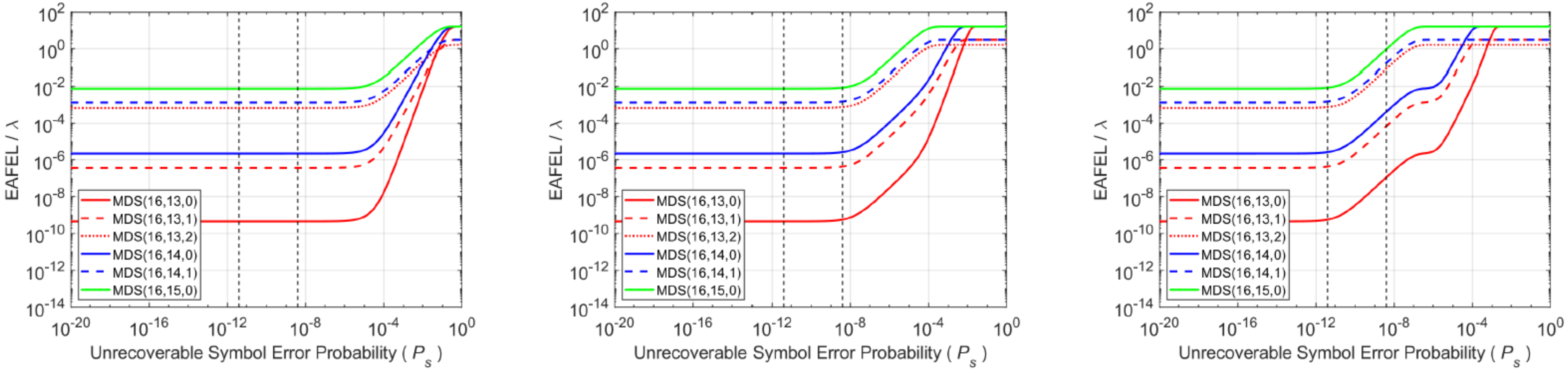
(a) Entity Size: 6.656 KB

(b) Entity Size: 6.656 MB

(c) Entity Size: 6.656 GB

- EAFEL affected at high sector error probabilities
- For small entity sizes, EAFEL unaffected by the presence of latent errors in the region of practical interest
- For medium and large entity sizes, EAFEL degrades in the region of practical interest
  - The larger the entity size, the more pronounced the degradation
- Increasing the number of parities (reducing l) improves reliability by orders of magnitude
- Employing lazy rebuild degrades reliability by orders of magnitude
- The declustered placement scheme achieves a significantly lower EAFEL than the clustered one

Effect of Latent Errors on EAFEL for Clustered Placement



(a) Entity Size: 6.656 KB

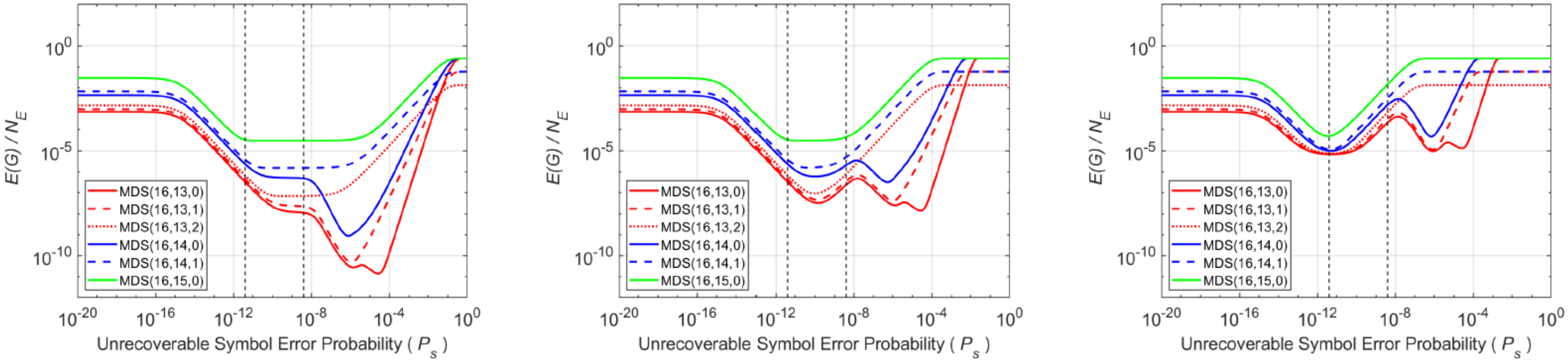
(b) Entity Size: 6.656 MB

(c) Entity Size: 6.656 GB

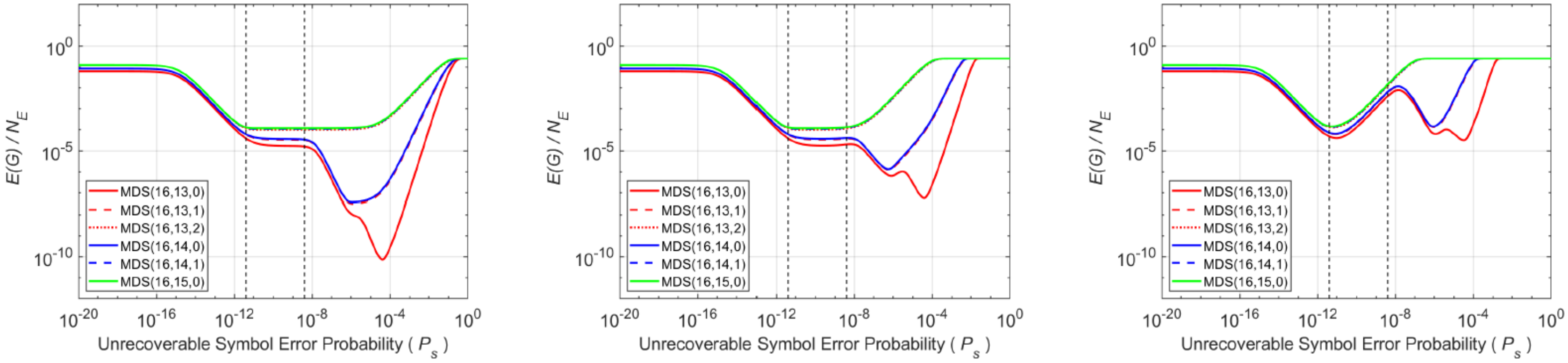
- EAFEL affected at high sector error probabilities
- For small and medium entity sizes, EAFEL unaffected by the presence of latent errors in the region of practical interest
- For large entity sizes, EAFEL degrades in the region of practical interest
  - The larger the entity size, the more pronounced the degradation
- Increasing the number of parities (reducing l) improves reliability by orders of magnitude
- Employing lazy rebuild degrades reliability by orders of magnitude
- The clustered placement scheme achieves a significantly higher EAFEL than the declustered one

Effect of Latent Errors on E(G)

Declustered Placement



Clustered Placement



(a) Entity Size: 6.656 KB (b) Entity Size: 6.656 MB (c) Entity Size: 6.656 GB

- In the interval [4.096×10^{-12}, 4.096×10^{-9}] of practical importance for P_s
  - E(G) significantly degraded by the presence of latent errors
  - E(G) not significantly affected by the employment of lazy rebuild

Summary

- Introduced the Expected Annual Fraction of Entity Loss (EAFEL) metric, which assesses the durability of distributed and cloud storage systems and reflects losses at an entity, say file or object, level
- Considered effect of the lazy rebuild scheme on the reliability of erasure-coded data storage systems
- Assessed the MTTDL and EAFEL reliability metrics using a non-Markovian analysis
- Derived closed-form expressions for the MTTDL and EAFEL metrics
- Demonstrated that system reliability is significantly degraded by the employment of the lazy rebuild scheme
- Established that the declustered placement scheme offers superior reliability in terms of both metrics
- Demonstrated that for practical values of unrecoverable sector error probabilities
  - MTTDL is adversely affected by the presence of latent errors
  - EAFEL is adversely affected by the presence of latent errors only when entities are large

Future Work

- EAFEL evaluation of erasure-coded systems for the cases of fixed-size entities that do not contain an integer number of codewords and of variable-size entities