Reducing Access Latency in Erasure Coded Cloud Storage with Local Block Migration

Y. Hu, D. Niu
University of Alberta
INFOCOMM 2016

Issues, Solutions, Issue

Content Availability → Redundancy Adding (Replication)

Storage Overhead → Erasure Coding

Access Efficiency → Systematic Coding

Server Load Balancing (Latency Tail) → ????

More on the Issues and Solutions

Lack of Real time Adjustment Flexibility → Optimize the Content Placement

Global shuffle is expensive.

Request is highly skewed and can vary.

→ Local Block Migration Solution for Content Placement Optimization and Control

Our Contributions

Local Block Migration Strategy

- A Statistical Model to Characterize the Load Balancing Problem
- Solution with Tricks to Keep coding consistency
- Method to Pick up the Best Local Move Efficiently at a time
- Theoretical Performance Results

Model

$$\text{(CP)} \quad \underset{y_1, y_2, \ldots, y_n}{\text{minimize}} \quad \mathbb{E}\left(\sum_{i=1}^{m} \frac{1}{2} L_i^2 \right) \tag{1}$$

$$\text{subject to} \quad L_i = \sum_{j: y_j = i} D_j, \quad \forall i, \tag{2}$$

$$y_i \neq y_j, \text{ if } G_i = G_j, \quad \forall i \neq j, \tag{3}$$

$$y_i = \{1, 2, \ldots, m\}, \quad \forall i, \tag{4}$$

Note that the mean and covariance for D can be estimated or measured.

Time slice based

N coded blocks

M servers

D: request for coded blocks in a time slice

L: server load in a time slice

y: Content Placement indicator

Transform into Min-K-Partitions

$$\mathbf{W} := \mathbb{E}(\vec{D} \cdot \vec{D}^T) = \vec{\mu} \cdot \vec{\mu}^T + \mathbf{\Sigma}, \tag{5}$$

$$\text{(CMKP+)} \underset{y_1, y_2, \ldots, y_n}{\text{minimize}} \quad \sum_{i<j} \mathbf{W}_{ij} \delta(y_i - y_j) + \frac{1}{2} \sum_{i} \mathbf{W}_{ii}, \tag{6}$$

$$\text{subject to} \quad y_i \neq y_j, \text{ if } G_i = G_j, \forall i \neq j, \tag{7}$$

$$y_i = \{1, 2, \ldots, m\}, \forall i, \tag{8}$$

$$\delta(x) := \begin{cases} 1, & \text{if } x = 0, \\ 0, & \text{otherwise.} \end{cases}$$

Solution

Penalize the weights of links that violate the constraint.

$$\mathbf{W}'_{ij} = \begin{cases} f_{ij}(\mathbf{W}) & \text{, if } G_i = G_j, i \neq j, \\ \mathbf{W}_{ij}, & \text{otherwise,} \end{cases} \quad (9)$$

Carry out the move which reduces the object the most at a time.

Update affected weights for next move.

Application

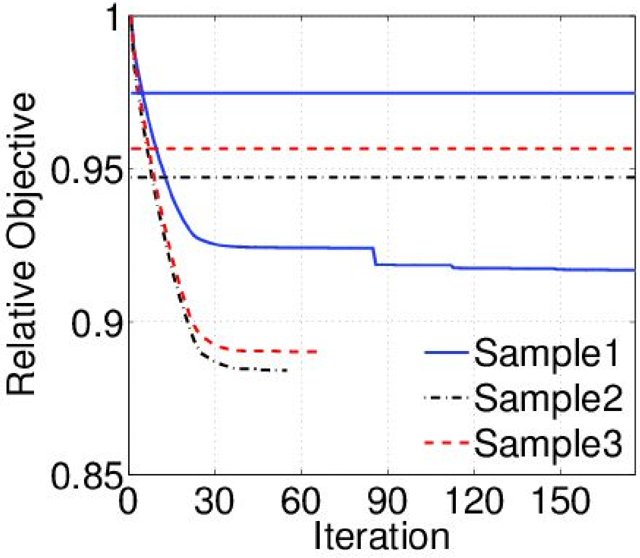
Static optimization for the Content Placement

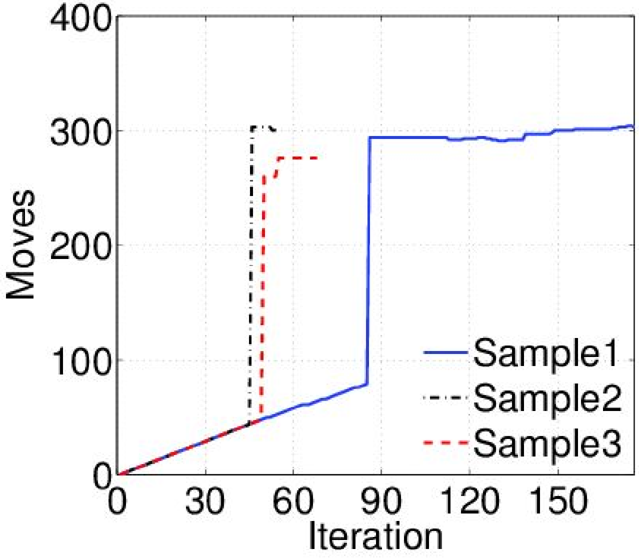
1. Collect the request statistics
2. Use the local block migration algorithm to get a placement strategy
3. Carry out the placement

Content Placement Optimization Control

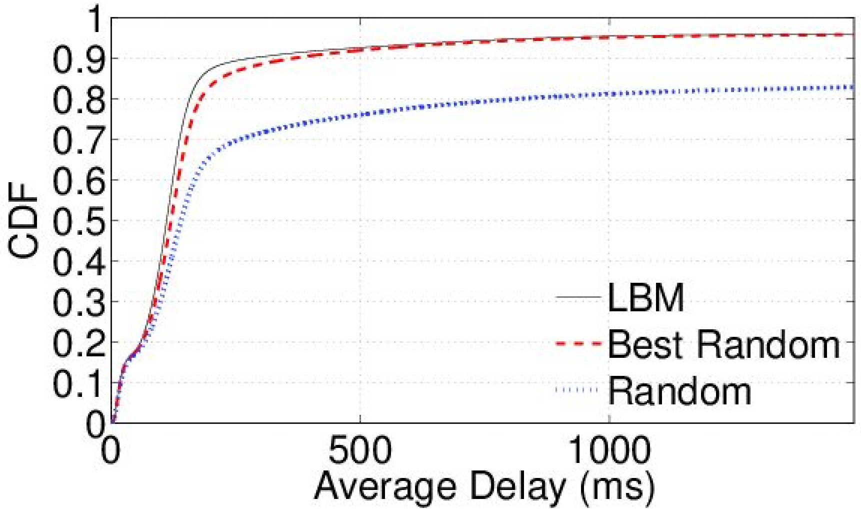
1. Collect the request statistics the past time window and use it as an estimation for the future time
2. Use the local block migration algorithm to decide some local best migrations
3. Carry out the local migrations
4. Wait for some time and goto 1.

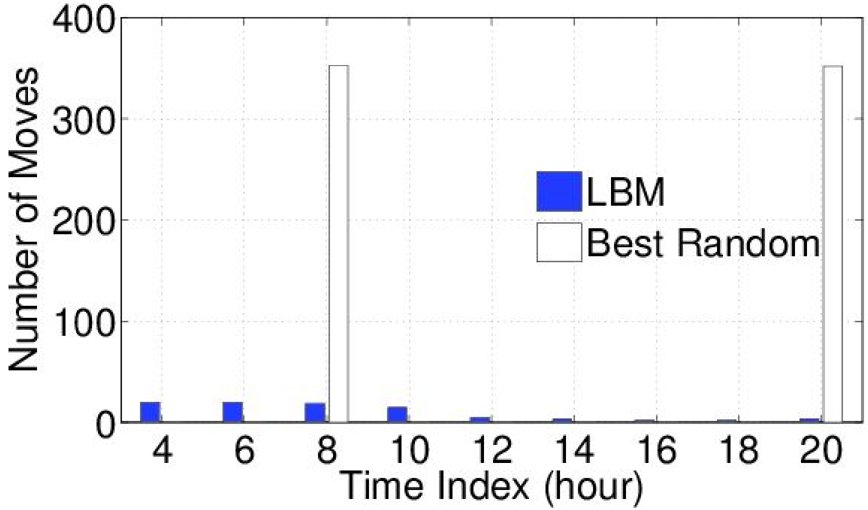
Simulation----Placement Optimization





Simulation----Optimal Control





Theoretical Results

1. Performance is bounded by

$$1 + \frac{1}{m - \alpha + 1}\left(\frac{\mathbb{E}\left(\left(\sum_i D_i\right)^2\right)}{\sum_i \mathbb{E}(D_i^2)} - 1\right).$$

2. Coding consistency is guaranteed for provided penalizing functions.

Questions.

Thank you so much!