Reconstruction and Repair Degree of Fractional Repetition Codes

Krishna Gopal Benerjee  
DA-IICT Gandhinagar  
Email: krishna_gopal@daiict.ac.in

Manish K. Gupta  
DA-IICT Gandhinagar  
Email: mankg@computer.org

Nikhil Agrawal  
DA-IICT Gandhinagar  
Email: nikhil_agrawal@daiict.ac.in

Abstract—Given a Fractional Repetition (FR) code, finding the reconstruction and repair degree in a Distributed Storage Systems (DSS) is an important problem. In this work, we present algorithms for computing the reconstruction and repair degree of FR Codes.

I. INTRODUCTION

We consider Distributed Storage Systems (DSSs) that use Distributed Replication-based Simple Storage (DRESS) codes consisting of an inner Fractional Repetition (FR) code and an outer Maximum Distance Separable (MDS) code to optimize various parameters of DSS [1], [2]. Constructing a FR code C(n, θ, α, ρ) for a given (n, k, d) DSS is an important problem addressed in [1]. On the other hand, given a FR code finding the reconstruction degree k (minimum number of nodes that needs to be contacted to reconstruct the entire data) and the repair degree d (number of nodes needs to be contacted in case of failure of a node) in such a (n, k, d) DSS has not been studied. Towards this end, for a given FR code, we define a reconstruction degree k̄ as the smallest number of nodes one has to contact to recover the entire file. This gives us a lower bound on actual reconstruction degree kFR defined as a degree such that user can get the entire data by contacting any (minimal) kFR nodes. For weak dress codes [2], finding the repair degree is non-trivial problem. Algorithm 2 computes it.

II. ALGORITHMS

Let the column support (non-zero entries) of a column Mj, 1 ≤ j ≤ θ of incidence matrix M of a FR code C(n, θ, α, ρ) [3] is Hj and the repair degree of each node Ui is d Ui.

Example 1. Consider a FR Code C(5, 9, 4, 2) (see [2]) with 9 packets and 5 nodes given as U1 = {1, 2, 3, 4}, U2 = {1, 6, 9}, U3 = {2, 5, 7, 9}, U4 = {3, 5, 6, 8} & U5 = {4, 7, 8}. Algorithm 1 gives k̄ = 3 and using algorithm 2, one finds the repair degree of node U2 = d2 = 3.

REFERENCES


An extended abstract of the paper can be obtained from Arxiv or author’s home page at http://www.guptalab.org

Algorithm 1 Algorithm to compute reconstruction degree k̄

Require: Node packet distribution of FR code after removing the last packet θ (as it can be recovered by parity) from all n nodes of Vn = {V1, V2, ..., Vn}.

Ensure: k̄ = Reconstruction degree

1: For 1 ≤ i, j, m ≤ n, if ∃ Vi & Vj s.t. Vj ⊆ Vi then delete all such Vj for all possible nodes Vi and list remaining collection of nodes as Vm = {V1, V2, ..., Vm}, |Vi| = αi = number of packets in node Vi.

2: Let Vl = {Vj, Vj ∈ Vm|1 ≤ j ≤ m & |Vj| = max{αj}}.

3: Pick an arbitrary set Vi ∈ Vl, and call this set as P. Set the counter kλ = 1, 1 ≤ kλ ≤ m and 1 ≤ λ ≤ |Vl| = l.

4: If ∃ Vi,j(1 ≤ j ≤ m) ∈ Vm s.t. Vi,j ∩ P = φ then go to step 5 otherwise jump to step 6.

5: Pick Vi,j(1 ≤ j ≤ m) ∈ Vm which has max cardinality among all Vi,j in Vm with Vi,j ∩ P = φ. Update P = P ∪ Vi,j, update counter kλ = (kλ + 1) and go to step 4.

6: If ∃ Vi(1 ≤ r ≤ m) ∈ Vm s.t. Vi ∉ P then go to step 7 otherwise go to step 8.

7: Pick Vi(1 ≤ r ≤ m) ∈ Vm which has maximum |Vi,j| s.t. all Vi,j ∈ Vm having the condition Vi,j ∉ P then update P = P ∪ Vi,j, update counter kλ = (kλ + 1) and go to step 6.

8: If 1 ≤ λ < l, then store kλ in k̄ and set k̄ = k(λ+1) and perform step 4 for P = Vi,j(1 ≤ j ≤ m) ∈ Vm s.t. Vi,j ≠ Vi ∈ Vm, otherwise report k̄ = min{k̄ λ=1}.

Algorithm 2 Algorithm to compute Repair Degree di

Require: Incidence matrix Mnx×θ of FR code and Hj.

Ensure: Repair degree di for a node Ui, 1 ≤ i ≤ n.

1: For each node i, 1 ≤ i ≤ n let Si(i) = {Hj\{i}|i ∈ Hj, 1 ≤ j ≤ θ}. Set q = 1, 1 ≤ q ≤ n.

2: Compute T ⊆ {1, 2, ..., θ} s.t. |T| > 1 is maximum among all possible subsets and for t ∈ T, Ht\{i} ⊆ Si(i), and \( \bigcap_{t} \{ Ht\{i} \neq \phi \). Set counter l(q) = 1 ≤ q ≤ n = |T| - 1. Store l(q) in l(q).

3: Update Si(i) = Si(i)\{Ht\{i\}, ∀t ∈ T.

4: If Si(i) = φ or singleton set or Ht\{i} ∩ Hs\{i} ∈ Si(i) = φ ∀1 ≤ r, s ≤ n then di = αi - ∑q=1 l(q), where αi = |Vi|, otherwise set q = q + 1 and go to step 2.