

**Network-Aware Feasible Repairs for Erasure-Coded Storage**

**Abstract:**

A significant amount of research on using erasure coding for distributed storage has focused on reducing the amount of data that needs to be transferred to replace failed nodes. This continues to be an active topic as the introduction of faster storage devices looks to put an even greater strain on the network. However, with a few notable exceptions, most published work assumes a flat, static network topology between the nodes of the system. We propose a general framework to find the lowest cost feasible repairs in a more realistic, heterogeneous and dynamic network, and examine how the number of repair strategies to consider can be reduced for three distinct erasure codes. We devote a significant part of the paper to determining the set of feasible repairs for random linear network coding (RLNC) and describe a system of efficient checks using techniques from the arsenal of dynamic programming. Our solution involves decomposing the problem into smaller steps, memorizing, and then reusing intermediate results. All computationally intensive operations are performed prior to the failure of a node to ensure that the repair can start with minimal delay, based on up-todate network information. We show that all three codes benefit from being network aware and find that the extra computations required for RLNC can be reduced to a viable level for a wide range of parameter values.

**Existing System:**

Traditional distributed storage systems employ replication to ensure data resilience, erasure coding provides equivalent or better resilience while using a fraction of the raw storage capacity required for replication. Both new technologies and increasing storage volume requirements suggest that erasure coding will continue to increase in importance as a factor in data center design. Offloading of encode and decode operations to GPUs, FPGAs and/or use of modern software libraries such as ISA-L, jerasure and Kodo promises to lower the computation costs of these operations, potentially expanding the set of cost-effective use cases for erasure coded storage. Additionally, the increased IOP

density and IO bandwidth of next–generation storage devices, such as NVMe (Non-Volatile Memory Express), as compared with rotating media or earlier SSD devices, promises to lower the effective IO costs associated with coded storage, further expanding the set of use cases.

**Proposed System:**

We propose a novel mechanism to reduce the number of computations based on Gaussian elimination. This is critical in enabling the checks to be performed on data distributions with a large number of storage nodes and files split into many fragments. Beyond reducing the computational load on the system, it plays an important role in our proposed framework by decreasing the time necessary to find a set of feasible repairs.

Thus, it limits the probability that a potentially unchecked repair strategy is used or alternatively that the repair is delayed to wait for the checks to finish. We base our solution on the realization that the matrices that need to be checked share many of their rows. Therefore, we propose reusing submatrices after they have been reduced to an upper triangular form and memoized, a common technique of dynamic programming. Larger matrices can then be built by merging smaller matrices prepared in advance.

Furthermore, many of these smaller matrices can also be reused in subsequent generations of failure and repair, further decreasing computational costs over the lifetime of the system. While our work is motivated by the challenges posed by making RLNC usable in environments such as data centers, it is applicable for other erasure codes that employ functional repair.