International Journal of Research in Advent Technology (IJRAT) Special Issue
E-ISSN: 2321-9637
Sharadchandra Pawar college of Engineering, Dumbarwadi, Pune 410504, Organizes
National Conference "MOMENTUM-17", 14th & 15th February 2017
Available online at www.ijrat.org

Performance Improvement of Hadoop Distributed File System Using Cauchy Coding Approach

Supriya Datkhile[1], Tanvi Padekar[2],

1,Visiting lecturer at GCOEARA, Department Of Computer Engineering, Awasari(KH)

2, Visiting lecturer at GCOEARA, Department Of Computer Engineering, Awasari(KH)

Email: supriyadatkhile@gmail.com[1]

Abstract-Clients of cloud storage for the most part dole out various repetition designs of eradication codes, contingent upon the wanted harmony amongst execution and adaptation to internal failure. Our study finds that with low likelihood, one coding plan picked by dependable guidelines, for a given repetition arrangement, performs best. In this paper, proposed CaCo, a proficient Cauchy coding approach for information storage in the cloud. To begin with, CaCo utilizes Cauchy matrix heuristics to deliver a matrix set. Second, for each matrix in this set, CaCo utilizes XOR schedule heuristics to produce a progression of schedules. At long last, CaCo chooses the most limited one from all the delivered schedules. In such a way, CaCo can distinguish an ideal coding plan, inside the ability of the current best in class, for a self-assertive given excess arrangement. By influence of CaCo's tendency of straightforwardness to parallelize, we support altogether the execution of the determination procedure with rich computational assets in the cloud. CaCo in the Hadoop disseminated record framework and assess its execution by contrasting and "Hadoop-EC" created by Microsoft inquire about. Our trial comes about demonstrate that CaCo can get an ideal coding plan inside worthy time.

Index Terms-Cauchy Matrix; Cloud Storage; Cauchy Coding (CaCo); Hadoop.

1. INTRODUCTION

Cloud stockpiling is developed of various modest and problematic segments, which prompts a reduction in the general mean time between failures (MTBF). As capacity frameworks develop in scale and are conveyed over more extensive systems, segment failures have been more basic, what's more, prerequisites for adaptation to non-critical failure have been further expanded. In this way, the disappointment assurance offered by the standard Attack levels has been no more adequate much of the time, what's more, stockpiling creators are thinking about how to endure bigger quantities of failures.

Cloud storage is built up of numerous inexpensive and unreliable components, which leads to a decrease in the overall MTBF (Mean Time between Failures). As storage system grow in scale and are deployed over wider networks, Components failure have been more common and requirements for fault tolerance have been further increased. So, the failure protection offered by the standard RAID levels has been no longer sufficient in many cases, and storage designers are considering how to tolerate larger numbers of failures. For ex. Google Cloud Storage, Windows Azure Storage, OceanStore, DiskReduce, Hail, and others all tolerate at least three failures.

CaCo can distinguish an ideal coding plan, inside the ability of the current best in class, for a self-assertive given excess arrangement. By influence of CaCo's tendency of straightforwardness to parallelize, system support altogether the execution of the determination procedure with rich computational assets in the cloud. System actualize CaCo in the Hadoop disseminated record framework and assess its execution by contrasting and "Hadoop-EC" created by Microsoft inquire about. System demonstrate that CaCo can get an ideal coding plan inside worthy time. Initially CaCo makes use of Cauchy matrix heuristics to generate a matrix set. Later for each matrix in the produced set, CaCo seeks XOR schedule heuristics to produce series of schedules. Lastly, CaCo chooses the shortest one from all the generated schedules. In this way for an arbitrary given redundancy configurations CaCo has capability to identify an optimal coding scheme, within the ability of present state of art. By taking the advantage of CaCo such as easy to parallelize it can significantly increase the performance through the selection process with

enormous computational resources in the cloud based systems. System incorporate CaCo in Hadoop Distributed File System (HDFS) and estimate its performance by doing comparison with "Hadoop-EC" developed by Microsoft research.

The rest of the paper has been organized as: section 2 highlights the related work along with their limitations, section 3 discusses the proposed work of system. Section 4 followed by conclusion and references.

2. RELATED WORK

Distributed storage is created up of different less expensive and hazardous included substances, which winds up in a lower inside the common MTBF (construe time between disillusionments). As limit structures make in scale and are passed on over more broad frameworks, issue screw ups had been more fundamental, and necessities for adjustment to inner disappointment were in addition expanded. Along these lines, the failure security gave by the same old RAID levels has been as of now not satisfactory in various illustrations, and parking space originators are considering how to persevere through broad amounts of frustrations. Google's conveyed stockpiling, home windows Azure stockpiling, Ocean keep, and others all persevere through no under 3 screw ups. To persevere through extra frustrations than RAID, various limit structures use Reed-Solomon codes for adjustment to non-basic disappointment. Reed-Solomon coding has been round for quite a while, and has a true speculative foundation.

Various tries have been set out to get this point. At to begin with, individuals find that the thickness of a Cauchy matrix coordinates the amount of XORs. In light of this, a measure of work has attempted to plan codes with low thickness. Moreover, some lower limits have been construed on the thickness of MDS Cauchy lattices. Inside the bleeding edge best in class, the least demanding approach to discover most decreased thickness Cauchy cross sections is to distinguish most of the systems and pick the quality one. Given a redundancy setup (okay; m;w), the wide arrangement of grids is (2w k+m), which is truly exponential in okay and m. in this way, the detail method for the most capable network makes feel best for some little cases.

With this Cauchy shocking heuristic, we initially form a Cauchy framework insinuated as GM. By then seclude (described over Galois field) each unobtrusive component of GM together with in section j by GM0;j , such that GM is redesigned and the components of line 0 are each of the "1". Inside the loosening up of the lines, which joins line i, we number the grouping of ones, recorded as N. By then we isolate the segments of section i by GMi;j , and separately depend the measure of ones, implied as Nj (j [0; k − 1]). in the long run, select the base from N;N0; : ;Nk−1 and perform the operations that create it. In like manner we succeed in building a system using Cauchy charming heuristic. The above two heuristics can convey a twofold matrix which joins less ones; in any case, it can never again be a complete one in the different Cauchy cross sections. The examination while in travel to decrease the measure of XOR operations inside the method for destruction coding has revealed that the wide arrangement of ones in a Cauchy cross section has lower limits. In this way, best by strategy for cutting down the thickness of the Cauchy grid, it's far difficult to improve the encoding general execution in a general sense.

Highly available cloud storage is often implemented with complex, multi-tiered distributed systems built on top of clusters of commodity servers and disk drives. Sophisticated management, load balancing and recovery techniques are needed to achieve high performance and availability amidst an abundance of failure sources that include software, hardware, network connectivity, and power issues [1]. Windows Azure Storage (WAS) is a cloud storage system that provides customers the ability to store seemingly limitless amounts of data for any duration of time. WAS customers have access to their data from anywhere at any time and only pay for what they use and store.e presented QBUiC, a query URL bipartite graph based approach to query recommendation.

Query recommendation system can adaptively recommend related queries to a given query by analyzing the query-URL history, consisting of three phases, i.e, preparation, graph generating, and HAC-based ranking phases [2]. MDS erasure codes are ubiquitous in storage systems that must tolerate

failures. While classic Reed-Solomon codes can provide a general-purpose MDS code for any situation, systems that require high performance rely on special-purpose codes that employ the bitwise exclusive-or (XOR) operation, and may be expressed in terms of a binary generator matrix [3].
System present longest lowest-density MDS codes, a simple kind of multi-erasure array code with optimal redundancy and minimum update penalty [4]. B. Calder present system succeed in building a system using Cauchy charming heuristic. The above two heuristics can convey a twofold matrix which joins less ones; in any case, it can never again be a complete one in the different Cauchy cross sections [5]. Krishna, Vibha examine while in travel to decrease the measure of XOR operations inside the method for destruction coding has revealed that the wide arrangement of ones in a Cauchy cross section has lower limits. In this way, best by strategy for cutting down the thickness of the Cauchy grid, it's far difficult to improve the encoding general execution in a general sense [6]. Chuanyi Liu1, Xiaojian Liu and Lei Wan define A policy based de-duplication proxy scheme is proposed It suggests a policy-based de-duplication proxy scheme to enable different trust relations among cloud storage components, de-duplication related components and different security requirements. Further proposes a key management mechanism to access and decrypt the shared de-duplicated data chunks based on Proxy Re-encryption algorithms. System finally analyses the security of the scheme.

3. PROPOSED WORK

A proposed system called CaCo- A Cauchy Coding Approach for Cloud Storage System. In Proposed System CaCo, a new approach that incorporates all existing matrix and schedule heuristics, and thus able to identify and optimal coding scheme within the capability of the current state of the at for a given redundancy configuration. The selection process of CaCo has an acceptable complexity and can be accelerated by parallel computing [7]. By influence of CaCo's tendency of straightforwardness to parallelize, system support altogether the execution of the determination procedure with rich computational assets in the cloud. The experimental results demonstrate that CaCo outperforms the "Hadoop-EC". Approach by 26.68- 40.18 percent in encoding time and by 38.4-52.83 percent in decoding time simultaneously.
The performance study analyzes their performance in terms of data encoding time and data decoding time.

1.Evaluation Methodology: Proposed system implement CaCo in HDFS and evaluate its performance of data encoding and decoding by comparing with Hadoop-EC.

2. Effectiveness of the Generated Schedules: The four Cauchy matrix heuristics used in the experiments are Cauchy Good, Optimizing Cauchy, Original, and Greedy.

3.Running Time of CaCo: For redundancy configuration, system run CaCo for three times and collect the average running time [8].

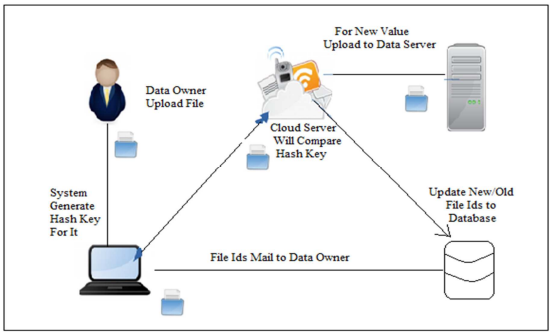

Fig. 1. System Architecture

3.1 System Overview:

The contribution of proposed system is two-fold. First, through a number of experiments and numerical analyses, we get some beneficial observations as follows:

1. Given a redundancy configuration ðk;m;wÞ, the shortest XOR schedule that one can get with a different Cauchy matrix has an obviously different size.
2. For a given redundancy configuration, there is a large gap in the coding performance when using different coding schemes.
3. None of existing coding schemes performs best for all redundancy configurations ðk;m;wÞ.

Second, based on the preceding observations, propose system CaCo, an efficient Cauchy Coding approach for cloud storage systems. CaCo uses Cauchy matrix

heuristics to produce a matrix set. Then, for each matrix in this set, CaCo uses XOR schedule heuristics to generate a series of schedules, and selects the shortest one from them. In this way, each matrix is attached with a locally optimal schedule [9],[10]. Finally, CaCo selects the globally optimal schedule from all the locally optimal schedules. This globally optimal schedule and its corresponding matrix will be stored and then used during data encoding and decoding. Incorporating all existing matrix and schedule heuristics, CaCo has the ability to identify an optimal coding scheme, within the capability of the current state of the art, for an arbitrary given redundancy configuration [11],[12].

System implement CaCo in the Hadoop distributed file system (HDFS) and evaluate its performance by comparing with "Hadoop- EC" developed by Microsoft research [13]. System experimental results indicate that CaCo can obtain an optimal coding scheme for an arbitrary redundancy configuration within an acceptable time. Furthermore, CaCo outperforms "Hadoop-EC" by 26.68-40.18 percent in the encoding time and by 38.4-52.83 percent in the decoding time simultaneously [14].

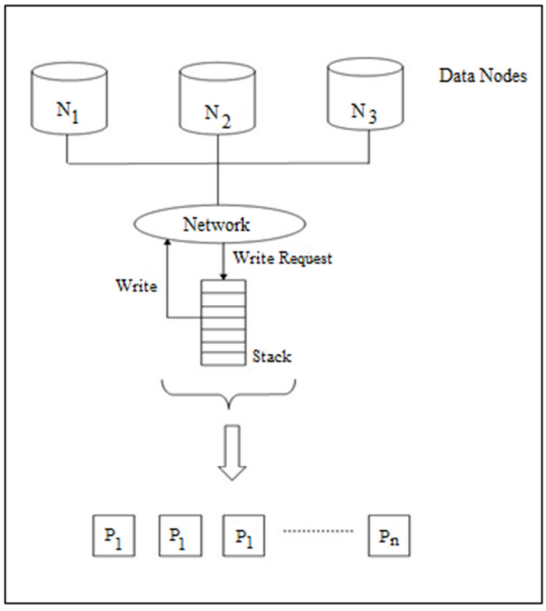

Fig. 2. System Structure (DataNodes)

4. PROPOSED ALGORITHM

4.1 Algorithm 1: Write Operation with CaCo

Input: Data Files for Process
Output: Store Data on Datanodes

Step 1: The Client sends a write request to the NameNode.
Step 2 : The NameNode allocates some DataNodes to the Client.
Step 3: Write the data blocks into DataNodes.
Step 4: Make a copy of data and put it into DataQueue.
Step 5: Encode data with the schedule selected by CaCo.
Step 6: Write the coding blocks into DataNodes.
Step 7: Data encoding finishes.
Step 8: Remove the copies of data from DataQueue.

5. CONCLUSION

In this propose system CaCo, a new approach that incorporates all existing matrix and schedule heuristics, and thus is able to identify an optimal coding scheme within the capability of the current state of the art for a given redundancy configuration. The selection process of CaCo has an acceptable complexity and can be accelerated by parallel computing. It should also be noticed that the selection process is once for all. System implement CaCo in the Hadoop distributed file system (HDFS) and evaluate its performance by comparing with "Hadoop- EC" developed by Microsoft research. CaCo can distinguish an ideal coding plan, inside the ability of the current best in class, for a self-assertive given excess arrangement. CaCo outperforms the "Hadoop-EC" approach by 26.68-40.18 percent in encoding time and by 38.4-52.83 percent in decoding time simultaneously. In future, the Cauchy's rule can be incubated with bandwidth of performance with respect to time for series computation of delay in network node retiring. The delay time reduction under narrow network is still a bottle neck situation. Either of this can be improvised with technical reduction of data sets and its indexing.

Acknowledgments

I express my sincere thanks to my project guide Prof. S. A. Kahate who always being with presence &

constant, constructive criticism to made this paper. I would also like to thank all the staff of Computer Department for their valuable guidance, suggestion and support through the paper work. Above all I express our deepest gratitude to all of them for their kind-hearted support which helped us a lot during project work. At the last I thankful to my friends, colleagues for the inspirational help provided to me through a paper work.

REFERENCES

[1] D. Ford, F. Labelle, F. I. Popovici, M. Stokely, V.-A. Truong, L. Barroso, C. Grimes, and S. Quinlan, "Availability in globally distributed storage systems," in Proc. 9th USENIX Symp. Oper. Syst. Des. Implementation, 2010, pp. 61–74.

[2] B. Calder, J. Wang, A. Ogus, N. Nilakantan, A. Skjolsvold, S. McKelvie, Y. Xu, S. Srivastav, J. Wu, H. Simitci, J. Haridas, C. Uddaraju, H. Khatri, A. Edwards, V. Bedekar, S. Mainali, R. Abbasi, A. Agarwal, M. F. u. Haq, M. I. u. Haq, D. Bhardwaj, S. Dayanand, A. Adusumilli, M. McNett, S. Sankaran, K. Manivannan, and L. Rigas, "Windows Azure storage: A highly available cloud storage service with strong consistency," in Proc. 23rd ACM Symp. Oper. Syst. Principles, New York, NY, USA, 2011, pp. 143–157.

[3] J. S. Plank, "XOR's, lower bounds and MDS codes for storage," Univ. of Tennessee, TN, USA, Tech. Rep. CS-11-672, May 2011.

[4] S. Lin, G. Wang, D. Stones, J. Liu, and X. Liu, "T-code: 3-erasure longest lowest-density mds codes," IEEE J. Sel. Areas Commun., vol. 28, no. 2, pp. 289–296, Feb. 2010.

[5] B. Calder, J. Wang, A. Ogus, N. Nilakantan, A. Skjolsvold, S. McKelvie, Y. Xu, S. Srivastav, J. Wu, H. Simitci, J. Haridas, C. Uddaraju, H. Khatri, A. Edwards, V. Bedekar, S. Mainali, R. Abbasi,"Windows azure storage: A highly available cloud storage service with strong consistency," in Proceedings of the Twenty-Third ACM Symposium on Operating SystemsPrinciples, SOSP '11, (New York, NY, USA), pp. 143–157, ACM, 2011.

[6] Krishna, Vibha V Dambal, B.N.Veerappa, "Bigdata Tolerance Optimization On Cloud Storage Systems", International Journal of Advance Research in Engineering, Science & Technology, Volume 3, Issue 6, June-2016.

[7] W. Gropp, E. Lusk, N. Doss, and A. Skjellum, "A highperformance, portable implementation of the MPI message passing interface standard," Parallel Comput., vol. 22, pp. 789–828, Sep. 1996.

[8] Ms.Krishna, B.N.Veerappa , "Bigdata Tolerance Optimization On Cloud Storage Systems ",International Journal of Advance Research in Engineering, Science & Technology, Volume 3, Issue 6, June-2016.

[9] J. Luo, L. Xu, and J. S. Plank, "An efficient xor-scheduling algorithm for erasure codes encoding," in Proc. IEEE/IFIP Int. Conf. Dependable Syst. Netw., 2009, pp. 504–513.

[10] J. L. Hafner, V. Deenadhayalan, K. K. Rao, and J. A. Tomlin, "Matrix methods for lost data reconstruction in erasure codes," in Proc. 4th Conf. USENIX Conf. File Storage Technol.-Volume 4, Berkeley, CA, USA, 2005, pp. 14–14.

[11] O. Khan, R. Burns, J. Plank, W. Pierce, and C. Huang, "Rethinking erasure codes for cloud file systems: Minimizing I/O for recovery and degraded reads," in Proc. 10TH USENIX Conf. File Storage Technol.,2012, pp. 251–264.

[12] K. M. Greenan, X. Li, and J. J. Wylie, "Flat xor-based erasure codes in storage systems: Constructions, efficient recovery, and tradeoffs," in Proc. IEEE 26th Symp. Mass Storage Syst. Technol., Washington, DC, USA, 2010, pp. 1–14.

[13] L. Xiang, Y. Xu, J. C. Lui, and Q. Chang, "Optimal recovery of single disk failure in RDP code storage systems," SIGMETRICS Perform. Eval. Rev., vol. 38, pp. 119–130, Jun. 2010.

[14] James S. Plank, Catherine D. Schuman, Devin Robison, "Heuristics for Optimizing Matrix-Based Erasure Codes for Fault-Tolerant Storage Systems," IEEE/IFIP - DSN 2012.