ISSN: 0975-8585

Research Journal of Pharmaceutical, Biological and Chemical Sciences

Map Reduce Implementation For Geo-Data Processing.

Revathy S*, Nivetha M, and Sahaya Dony Lency A.

Sathyabama University, Chennai, Tamil Nadu, India.

ABSTRACT

This paper proposes the cost minimization problem via a joint optimization of these three factors for big data services in geo-distributed data centers. Data placement, Task Assignment and Data Routing to describe the task completion time with the taken care of both data transmission and computation, we propose a Two-Dimensional Markov Chain and derive the average work completion time in closed-form. Furthermore, we model the problem as a Mixed-Integer Non-Linear Programming (MINLP) and taken as an efficient solution to linearize it.The optimized Data Center Resizing (DCR) is used to decrease the computation cost with the help of Map Reduce by adjusting the number of activated servers via Task Placement.

Keywords: Map Reduce, Big Data, Data Placement, Data Flow, Distributed Data Centers, Cost Minimization, Task Assignment

Corresponding author

ISSN: 0975-8585

INTRODUCTION

The existing routing strategy among data centers break to exploit the link diversity of data center networks. Due to the storage and calculation capacity constraints, not all tasks can be placed onto the same server, on which their corresponding data reside. It is un-prevented that certain data must be downloaded from a remote server. In this case, routing strategy matters on the spread cost. As indicated by Jin et al., the transmission cost, e.g., energy, nearly proportional to the number of network link used. Data locality may result in a waste of resources. Data locality may result in a waste of resources. The most computation resource of a server with less popular data may stay idle. The low resource utility further causes more servers to be operated and hence higher operating cost. Data locating and data movement and data computation more time. Big data refers to high volume, high velocity, and or high variety information assets that require new types of processing to enable improved decision making, insight discovery and process optimization. Due to its huge volume and complexity, it leads to difficult to process big data using on-hand database management tools. An effective option is to store huge data in the cloud, as the cloud has capabilities of storing big data and processing huge volume of user access requests in an achievable way. When hosting big data into the cloud, the data security becomes a major one as cloud servers cannot be fully believed by data owners.

The policy updating is a difficult issue in attribute-based access control systems, since once the data owner outsourced data into the cloud, it would not keep a copy in local systems.

When the data owner needs to change the access policy, it has to transfer the data back to the local site from the cloud, re-encrypt the information under the new access policy, and then move it back to the cloud server. By doing so, it incurs a big communication overhead and heavy computation burden on data owners. This motivates us to enhance a new method to outsource the task of policy updating to cloud server.

The grand challenge of outsourcing policy adding to the cloud is to guarantee the following requirements:

1. Correctness: Users who possess ample attributes should still be able to decrypt the data encrypted under new access policy by running the actual decryption algorithm.
2. Completeness: The policy including method should be able to update any type of access strategy.
3. Security: The policy updating should not overcome the security of the access control system or launch any new security problems.

The contributions of this paper include:

1. We regulate the policy adding problem in ABEsystems and design a new method to outsource the policy adding to the server.
2. We launch an eloquent and efficient data access control strategy for big data, which enables capable dynamic policy updating.
3. We design policy updating algorithms for different categories of access policies, e.g., Boolean Formulas, LSSS Structure and Access Tree.

Juxtapose to the conference volume, we too put forward an efficient and secure policy checking method that enables data owners to analyze whether the cipher texts have been updated rightly by cloud server. In this way, we do not need any help of data receiver, and data sender can check the correctness of the ciphertext updating by their own secret keys and checking keys presented by each authority [3]. Our method can also guarantee senders cannot use their confidential keys to decrypt any cipher texts encrypted by other data senders, although their secret keys have the components related with all the attributes. Moreover, we converse about some key features of the attribute-based control scheme and show how it is acceptable for big data access control in the cloud. What's more, we also include more process implification on policy updating algorithms and the policy examines method.

LITERATURE SURVEY

Big Data contains large-volume, complex and growing data sets with many, autonomous sources. Big data processing is the explosive growth of request on computation, storage, and communication in data centers, which unpleasant considerable operational outlay to data center providers. Therefore, to reduce the

cost is one of the problems for the upcoming big data era. Using these three components, i.e., task assignment, data placement and data routing, far down influenced by the operational outlay of geo distributed data centers. Here, we are ambitious to learn the cost minimization problem through a joint optimization [**1**] of these three factors for huge data making in geo-distributed data centers. Proposed using n-D Marko chain [**2**] and procure average task completion time.

The popularity and bandwidth usage of cloud services has increased rapidly in recent years. To provide users cloud storage with less synchronization latency, cloud storage donor are interested whether the achievement of their datacenter topology is effective for their users and how they can improve it. It is not understandable whether distributed cloud storage data-center topologies perform good centralized ones. In this paper, a comparison between centralized and spread cloud storage data-center topologies is made [**3**]. The topologies used by disguised cloud storage applications are analyzed with data together at global vantage points. The average amount is used as performance criteria. The result of this paper implies that using a distributed data-center topology has an affirmative effect on average throughput compared to a centralized topology. This research consider in getting an sensitive of the impact of various cloud storage [**4**] data-center topologies on the performance undergo by cloud storage users.

Our aim is to attain an optimal tradeoff between energy efficiency and service accomplishment over a set of distributed IDCs with constant change demand. In particular, we consider the outage probability as the Qi's metric, where power failure is defined as service demand over limiting the capacity of an IDC [**5**]. Our goal is thus to minimize total intensity cost over all IDCs, subject to the outage probability dependencies. We achieve the goal by dynamically changing server capacity and performing load changing in different time scales. We propose 3various load shifting[**6**] and joint capacity assignment schemes with various difficulty and performance. Our schemes support both stochastic multiplexing obtain and electricity-rate diversity [**7**].

BACKGROUND & RELATED WORKS

Markov chain

A Markov is a mathematical system that undergoes transitions from one state to another on a state space. It is an odd process generally characterized as memory-less[**8**]: the next state depends only on the current state and not on the chain of events that prepared it. Memorylessness is called the Markov property. Markov chains have more applications[**9-11**] as graphical models of real-world processes. A Markov chain is a sequence of odd variables X1, X2, X3, with the Markov property, namely that, shows the present state, the future and past states are not dependent[**12**]. Formally, if both dependent probabilities are well explained, i.e. if the possible values of Xi form a pre-determined set S called the state space of the chain. I-D Markov chain [**13**]works lies upon the processing, loading and task distribution.

Data Placement

The information placement is another huge issue in the geo-distributed data centers[**14**].Because where the data's are placed in the servers and how they can be approaches and compute the latency time of that particular data transition and move information data to the closest datacenter. However, the easy heuristic ignores 2 major sources of rate to datacenter operators: WAN bandwidth [**15**] between data centers, and more-provisioning datacenter capacity to accept highly skewed datacenter utilization. In this paper, we explain that a more luxurious approach can both dramatically decrease these rates and still further reduce user latency[**16**]. Proposed using Volley algorithm for automated info placement in geo distributed data centers. Once the data can be generated means abundant to analyze the migration data periodically.

Electricity cost

The electricity rate another burden in geo-distributed data centers [**17**]. Since more energy will be using in data centers. All the hardware's job without electricity says a novel, data-centric algorithm used to decreases energy rates and with the guarantee of heat-reliability[**18**] of the servers in geo distributed data centers[**19**].And also using the n-dimensional Markov chain algorithm to reduce the electricity cost.

ISSN: 0975-8585

Server cost

In geo distributed data centers hundreds of servers used. Because of this automatically the server cost will be more. How to decrease the server cost means using communications and data such as task assignment, data placement and data routing via an-D Markov chain. To efficiently directs the Datacenter resizing[**20-21**]. Proposed the optimal workload and balancing of latency, electricity prices and the energy consumption. Placement and task assignment approach. Number of sever will be reduced means at a mean time the energy cost also decrease. Server cost reduced using the joint optimization of these three factors.

PROPOSED SYSTEM

In this paper, we proposed to take care the cost minimization problem of big data processing with joint reward of data placement, task assignment and data routing, as shown in Fig 1. To describe the amount-constrained calculated and transmission in big data processing process, we put forward 2D Markov chain and derive the expected job completion time in closed form. Data Center Resizing (DCR) has been proposed to reduce the handling cost by changing the number of activated servers via task placement. We consider the problem as a Mixed-Integer Non-Linear Programming (MINLP) and put forward about an effective solution to linearize it. Data movement and Data computation time to be reduced. Cost for hardware expenditure is reduced. Number of servers reduced using task assignment process.

A. Two Dimensional Markov Chain
B. Mixed Integer Non-Linear Program
C. Data Center Record
D. Server Cost Minimization

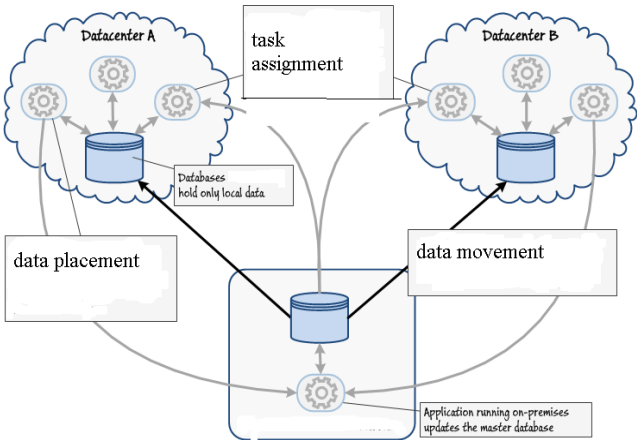

Fig 1: Proposed Architecture

Two Dimensional Markov Chain

A two-dimensional Markov chain and simplify the average task completion time in closed-form. The processing procedure then can be explained by a two-dimensional Markov chain, where each state represents pending tasks and available data chunks. The amount of computation resource that chunk occupies. The processing rate of task is proportional to its computation resource usage. We consider sufficient bandwidth on each link that can be handled as a fixed number, which is mainly determined by I/O and switch latency.

Mixed Integer Non-Linear Program

The big calculation difficulty of solving Mixed integer Non-linear program, we linearize it as a joint-integer linear programming problem, which can be simplified using commercial solver. Through extensive numerical studies, we display the huge efficiency of our proposed joint-optimization based procedure. The total energy cost then would be computed by adding up the rate on each server along all the geo-distributed data centers and the communication cost. The number of replicas for each data parts is a predetermined

constant. If it is a part of the optimization, denoted by an integer variable, the total cost can be further minimized.

Data Center Record

In data centers, each of which is with the exact number of servers. The data size, storage Requirement, and job arrival rate are all randomly generated. When the total number of servers increases, the communication costs of both algorithms reduce significantly. This is reason that more tasks and data chunks can be stored in the same data center when more servers are available in each data center. Hence, the communication cost is highly reduced. The communication costs of both algorithms converge and the reason is that most tasks and their mapping data chunks can be placed in the actual datacenter, or also in the actual server. Further increasing the number of servers will not affect the distributions of jobs or data chunks any more.

Server Cost Minimization

Large-scale data centers have been located all over the world furnishing services to 100s of 1000s of users. A data center may consist of large numbers of servers and ingest megawatts of power. Millions of amount on electricity cost have posed a heavy burden on the operating estimation to data center providers. Therefore, decreasing the electricity cost has received remarkable attention from both academia and industry. Among the methodologies that have been put forward so far for data center energy management, the techniques that covers lots of notice are task placement. The task placement is usually together considered to match the computing requirement.

DATA CENTER FORMATION

Data center formation is 1 of the sensitive models, switches, servers. In general, the transmission rate CR for inter-data center network is greater than CL for local transmission rate, i.e., CR > CL. Without loss of validity, all servers in the connection have the same computation expedient and storage capacity, both of which are normalized to 1 unit. We develop the data center with this ratio as shown in Fig 2. If the data center formation is very well means that data center is act fast. And it consumes low power when compare toordinary data centers.

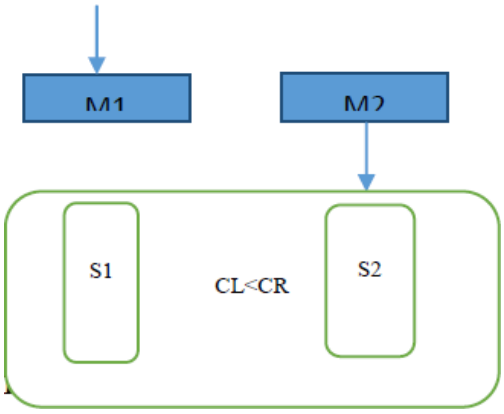

Fig 2: Data Center Formation

Data chunk means data can be further divided into number of small information's as shown in Fig 3.That information or information can be located into various servers. Its 1 of the simple path to associate the info from large set of data's in data center.

ISSN: 0975-8585

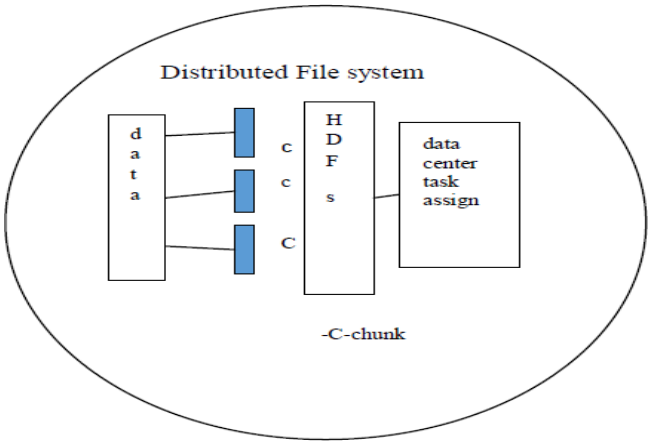

Fig 3: Data Chunk

Table 1: Notations Used

j_i	The set of servers in data center i
m_i	The switch in data center i
$\omega^{(u,v)}$	The weight of the link(u,v)
Φ_k	The size of the chunk k
λ_k	The task reaching rate for data chunk k
P	The number of data chunk replicas
D	The maximum anticipatedresponse time
P_j	The power utilization of server j
$\gamma^{(u,v)}$	The transmission rate of link(u,v)

Table 2: Variables Used

x_j	A binary variable indicating if server j is activated or not
y_{jk}	A binary variable pointing if chunk k is placed on server j or not
$z_{jk}^{(u,v)}$	A binary variable pointing if link (u,v) is used for flow for chunk k on server j
λ_{jk}	The request rate for chunk k on server j
θ_{jk}	The cpu usage of chunk k on server j
μ_{jk}	The cpu processing rate of chunk k on server j
$f_{i,k}^{(u,v)}$	The flow for chunk k destined to server j through link(u,v)

CONSTRAINTS OF DATA AND TASK PLACEMENT

We define a binary variable y_{jk}as shown in table 1in equation (1)to denote whether chunk k is located on server j as follows,

$$y_{jk} = \begin{cases} \textbf{1 if chunk k is placed on server j,} \\ \textbf{0 otherwise} \end{cases} \quad (1)$$

In the distributed file system, we look into equation (2) P copies for each chunk k < K, which simplifies to the following constraint:

$$\sum_{j \in J} yjk = P, \forall k \in K. \quad (2)$$

Furthermore, the data stored in each server j belongs to J cannot be more than its storage capacity as in equation (3), i.e.,

ISSN: 0975-8585

$$\sum_{k \in K} yjk.\, \emptyset k \leq 1, \forall j \in J. \qquad (3)$$

The data placement and task assignment are reveal to the data users with guaranteed QOS.

Let be the processing cost and loading cost for data chunk k on server j, correspondingly as in equation (4).The processing procedure then can be described by a 2D markov chain process. According to the QoS requirement $d_{jk} \leq D$,

$$\mu jk \quad \gamma jk - \lambda jk \geq \frac{Ujk}{D}, \forall j \in J, k \in K. \qquad (4)$$

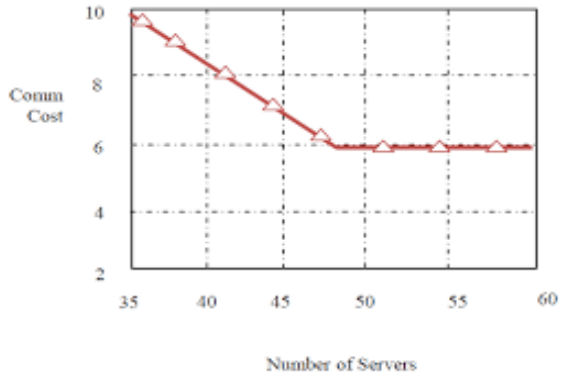

Fig 4: Cost Minimization on Effect of Servers

The binary variables of u_{jk} in (5)can be described by constraints

$$\Lambda_{jk} \leq u_{jk} \leq A\lambda_{jk}, \quad \forall j \in J, k \in K. \qquad (5)$$

Where A is an arbitrary large number as shown in table 2, because of $0<\lambda_{jk}<1$ and $u_{jk} \in \{0,1\}$.

Thus, when the number of servers gets increasing it will not affect the distributions of tasks or data chunks any more. Similar results are observed in Fig 4.

COST MINIMIZATION USING MAPREDUCE ALGORITHM

Numerous algorithms were defined earlier in the analysis of huge data set. Will go through the various work done to handle Big Data. In the beginning different algorithm was used previously to determine the big data. In work done by Hall. et al. there is defined a way for forming the rules of the large set of training data. The methodology is to have a one decision system generated from a huge and independent n subset of data. Here we use cost minimization using map reduce algorithm as follows, Cost Minimization using MapReduce Algorithms. Marking by S the set of input objects for the underlying problem. Let n, the problem cardinality, be the number of objects in S, and t be the number of devices apply in the system. Define m = n/t, namely, m is the number of objects per machine when S is equally distributed across the devices. Consider an algorithm for solving a problem on S. We say that the algorithm is minimal cost if it has all of the following traits.

• Minimum footprint: at all times, each machine uses only O(m) space of accumulations.
• Bounded net-traffic: in each round, every device sends and receives at most O(m) words of data over the network.

Constant round: the algorithm must stop after a number of rounds.

CONCLUSION

A two-dimensional Markov chain and simplify the average task completion time in closed-form. Furthermore, we model the problem as a collective-integer non-linear programming (MINLP) and points out an efficient solution to linearize it. The big efficiency of our proposal is validated by widened simulation based studies. Now it is currently undergoing with the weather report data sets. Only the study has been carried out in paper. The implementation will be given in later publication. This can also be done with the medical data sets.

REFERENCES

[1] Lin Gu, Deze Zeng, Peng Li and Song Guo. 12th International conference on communications 2012; 58-69.

[2] Kuangyu Zheng, Xiaodong Wang, Li, and Xiaorui Wang. Communications of the ACM 2010; 55(1): 167–176.

[3] Raghavendra R, Ranganathan P, Talwar V, Wang Z, and Zhu X. 13th International Conference on (ASPLOS). ACM 2008; 48–59.

[4] Liu Z, Lin M, Wierman A, Low SH, and Andrew LL. Proceedings of International Conference on Measurement and Modeling of ComputerS ystems (SIGMETRICS) ACM 2011; 233–244.

[5] Rao L, Liu X, Xie L, and Liu W. Proceedings of the 29th International Conference on Computer Communications (INFOCOM). IEEE 2010; 1–9.

[6] Hong Xu BL, Chen Feng. Proceedings ofInternational Conference on Measurement and Modeling of ComputerSystems (SIGMETRICS). ACM 2013; 33–36.

[7] Dean J and Ghemawat S. Communications of the ACM 2008;51(1): 107–113.

[8] Yazd SA, Venkatesan S, and Mittal N. SIGOPS Oper Syst Rev 2013;47(2): 33–40.

[9] Marshall I and Roadknight. Computer Networks and ISDN Systems 1998;30(223): 2123 – 2130.

[10] Jin H, Cheocherngngarn T, Levy D, Smith A, Pan D, Liu J, and Pissinou N. Proceedings of the 27th International Symposium on Parallel Distributed Processing (IPDPS)2013;623–634.

[11] Qureshi A, Weber R, Balakrishnan H, Guttag J, and Maggs B. Proceedings of the ACM Special Interest Group on Data Communication(SIGCOMM). ACM 2009;123–134.

[12] Fan X, Weber WD, and Barroso LA. Proceedings of the 34th Annual International Symposium on Computer Architecture (ISCA) ACM 2007; 13–23.

[13] Govindan S, Sivasubramaniam A, and Urgaonkar B. Proceedings of the 38th Annual International Symposium on Computer Architecture (ISCA) ACM 2011; 341–352.

[14] Gao PX, Curtis AR, Wong B, and Keshav S. Proceedings of the ACM Special Interest Group on Data Communication (SIGCOMM). ACM 2012; 211–222.

[15] Liu Z, Chen Y, Bash C, Wierman A, Gmach D, Wang Z, Marwah M and Hyser C. Proceedings of International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS) ACM 2012; 175–186.

[16] Sathiamoorthy M, Asteris M, Papailiopoulos D, Dimakis AG, Vadali R, Chen S, and Borthakur D. Proceedings of the 39th international conference on Very Large Data Bases, ser. PVLDB'13. VLDBEndowment 2013;325–336.

[17] Hu B, Carvalho N, Laera L, and Matsutsuka T. Proceedings of the 14th International Conference on Information Integration and Web-based Applications & Services, ser. IIWAS '12. ACM 2012; 167–176.

[18] Cohen J, Dolan B, Dunlap M, Hellerstein JM, and Welton C. Proc VLDB Endow 2009;2(2): 1481–1492.

[19] Kaushik R and Nahrstedt K. International Conference for High Performance Computing, Networking, Storage and Analysis (SC) 2012; 1–11.

[20] Chen F, Kodialam M, and Lakshman TV. Proceedings of the 29th International Conference on Computer Communications (INFOCOM). IEEE 2012;1143–1151.

[21] Shachnai H, Tamir G, and Tamir T. Theoretical Computer Science 2012;460: pp. 42–53.