International Journal of Database Theory and Application
Vol.8, No.3 (2015), pp.69-76
http://dx.doi.org/10.14257/ijdta.2015.8.3.07

Cloud-Enabled Data Center Organization using K-D Tree

Sandip Roy[1], Rajesh Bose[2], Tanaya Roy[3] and Debabrata Sarddar[4]
[1, 2, 4]University of Kalyani, Kalyani, West Bengal, India
[3]Brainware Group of Institutions, Kolkata, West Bengal, India
[1]sandiproy86@gmail.com,
[2]bose.raj00028@gmail.com,[3]tanaya.roy_2008@yahoo.co.in,
[4]dsarddar1@gmail.com

Abstract

An efficient load balancing algorithm in the field of cloud computing is absolutely essential in order to have a cloud network with graceful performance satisfying user expectation. Inspite of the existence of handful lightly loaded data centers, numerous heavily overloaded data centers may lead to a performance degradation of overall cloud network. Proper workload distribution may improve the overall performance of the cloud system. Now a day's eminent cloud division rules are highly demanding algorithm for distributing workloads among various cloud server nodes deployed in cloud-enabled data centers scattered over the geographical regions. For the researcher, cloud division rule and optimal cloud server node searching are the most demanding jobs in load balancing leading towards more efficient cloud network and improve users' satisfaction. This paper presents an expeditious cloud division rule based on geographical location of the cloud-enabled data centers distributed over earth surface and builds a two-dimensional space partition k-d tree to partition them in order to search intended cloud server node efficiently. Our proposed organization scheme can be utilized for active monitoring load balancing algorithm to improve the resource utilization for high performance in present cloud computing environment.

Keywords: *Cloud division rule, Cloud-enabled data center, k-d tree, Load balancing.*

1. Introduction

Cloud computing egresses as a new epitome of wide scale parallel and distributed computing in present IT world. Its revolutionary computing model assures increased flexibility, scalability and reliability with decreased operational and support cost as a utility. It counts the uncountable likes of business entrepreneurs and individuals by changing the dimension of conventional parallel and distributed computing. Cloud is a shared pool of configurable heterogeneous resources (e.g., networks, servers, storage, applications, and services) and mesh of vast on-demand infrastructure. Infrastructure not only refers to the application delivered to the end users as services pay-as-you-say-manner over the internet but also to the hardware and system software required for providing end users' required services. Computation in cloud encompasses maximum resource utilization with higher availability at minimized cost.

A cloud network consists of several distributed servers deployed in geographically dispersed cloud-enabled data center which is referred as cloud node in this context. All cloud nodes can be totally demarcated over the earth surface with respect to their geographic coordinate such that each of them can be represented by an ordered tuple of latitude and longitude respectively. As user request pattern is uncertain and cloud nodes differ in their current working status,

ISSN: 2005-4270 IJDTA

load balancing is a key phenomenon for improving overall system performance and greater stability of cloud system. Load balancing is the process of distributing the work load among various cloud nodes of a cloud system for ameliorating both resource utilization and job response time while minimizing the over loaded cloud node counts [1]. Load balancing is a challenging task for the researchers to optimize resource utilization, maximizing the system throughput, minimize response time to avoid the inconvenient situation of the cloud system. Typically load balancing algorithms can be of two types, static and dynamic algorithms. In case of static load balancing algorithm prior knowledge of the cloud system is needed and does not depend on the current cloud system status. On the other hand, dynamic load balancing algorithm is based on the current cloud system status and its' performance is better than the static algorithm [1]. A suitable cloud division rule may smooth the task of load balancing expeditiously. In this article our goal is to develop a novel cloud division rule over public cloud, consists of numerous cloud nodes, based on the geographical coordinates that can be used to dynamically distribute the workload among cloud nodes [1, 2]. Formulating this efficient cloud division rule phenomenon, a concept of space partitioning data structure in a two-dimensional space for organizing cloud nodes over the earth surface is introduced. This proposed cloud division rule logically partitions cloud nodes over the earth surface that can be used to simplify the different load balancing approaches based on cloud division rule. In cloud system an index table containing necessary information of all registered cloud node is maintained. When a cloud system comprises of numerous number of cloud nodes, then it is a time consuming procedure to search an intended cloud node from that index table. Usage of our proposed organizing structure could lead to lesser time consuming searching procedure. So the task of assigning a new arrival job using efficient load balancing approach to proper cloud node in the cloud system could be handled in more effective way.

2. Related Works

In this section few literatures that substantiate the importance of load balancing and cloud division rules are discussed. Ram Prasad Padhy and P Goutam Prasad Rao talked about the fundamental concepts of cloud computing and different load balancing approaches in cloud computing system to achieve maximum resource utilization [3]. Gaochao Xu, Junjie Pang and Xiaodong Fu have proposed load balancing model based on cloud partitioning in order to design public cloud model with better performance. P. Jamuna and R. Anand Kumar have introduced an optimized cloud partitioning technique for simplifying load balancing based on cloud clustering technique [4]. In [5 - 11] various authors explained various load balancing algorithms which deal with different load balancing problems in cloud environment which are used to distribute their workload over multiple cloud nodes in order to achieve their objective, i.e. the maximizing the system throughput, to avoid the overloading. Good load balancing makes cloud computing more efficient and also improves user satisfaction [12]. Thus formulation of a good cloud partition rule helps innovative load balancing approaches to reach towards an efficient and powerful cloud system. Idea of our proposed system model is to construct a two-dimensional space partitioning tree that could enable dynamically balancing the cloud workloads among the cloud nodes with different load degree (ideal, normal or overloaded).

Section 3 exposits our proposed cloud division rule using the concept of k-d tree for partitioning cloud-enabled data center over the earth surface and section 4 encloses a conceptual example of our proposed cloud system organization. At long last conclusion is explained in section 5.

3. Our Proposed Cloud Division Methodology

A k-d tree is a space partitioning data structure that organizes points eminently in a k-dimensional space. k-d trees are a special case of binary space partitioning trees that is useful for searches involving multidimensional search key such as nearest neighbor search. A cloud node in a cluster may be far from other nodes or cluster in same geographic region may still far apart. So a detailed cloud division methodology needs to be formed in such a manner that an intended cloud node can be accessed very efficiently. Thus formulation of such cloud division rule results in a hierarchical structure that can be implemented using k-d tree is a novel idea.

Each cloud node can be represented using their geographic coordinates (i.e. latitude and longitude) over the earth surface. Initially in our proposed model the set of cloud nodes are partitioned based upon Equator and Prime meridian i.e. into four major quadrants viz. North-East, North-West, South-East and South-West as shown in Figure 1. After initial partitioning each partition can be further sub partitioned (sub-sub partitioned) by two-dimensional space partitioning algorithm [13]. Our cloud division approach based on the idea of k-d tree is articulated below.

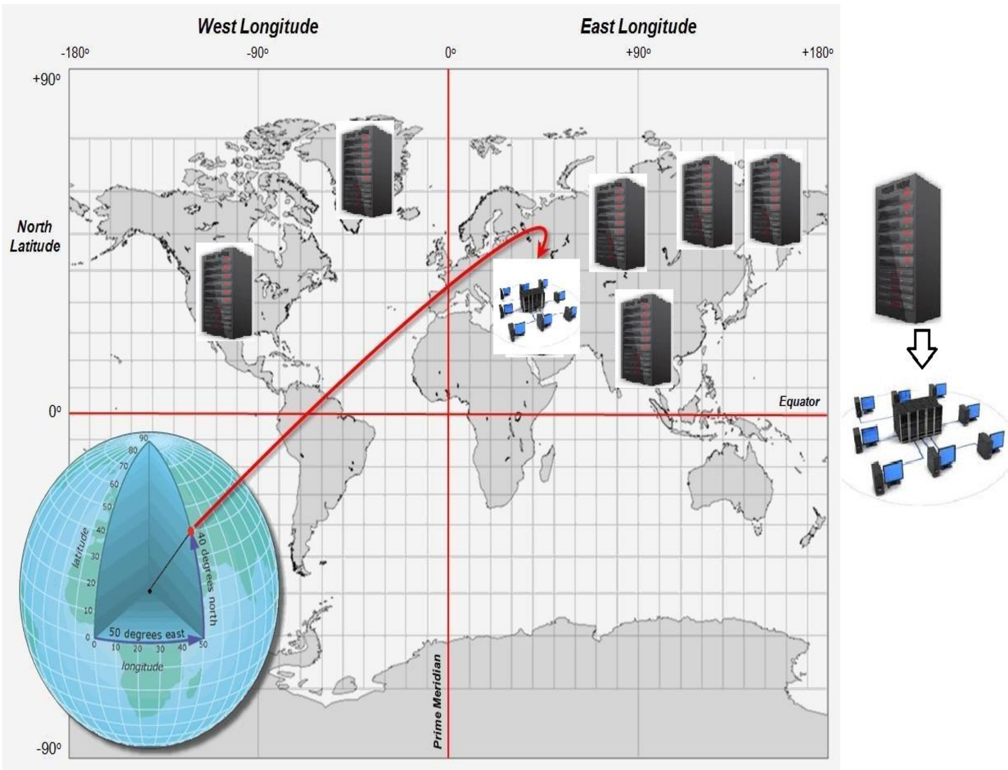

Figure 1. Geographically Dispersed Cloud-enabled Data Center

3.1. Algorithm 1 Proposed Cloud Division Rule

Input: Set of latitudes and longitudes of registered cloud node

Output: Two-dimensional space partition tree demarcating all cloud nodes

1. Call Algorithm 1.1 to choose proper quadrant
2. Call algorithm 1.2 to construct k-d tree for selected quadrant

Algorithm 1.1 Selection of proper quadrant

1. For each cloud node
1.1 Set the value x by latitude angle of cloud node
1.2 Set the value y by longitude angle of cloud node
1.3 Set the value DirLat by the direction latitude (North or South) of cloud node
1.4 Set the value DirLong by the direction longitude (East or West)of cloud node
1.5 Initially, total geographical space is partitioned into four quadrants based on Equator and Prime meridian (e.g. North-East, South-East, South-West and North-West). Place the cloud node at proper quadrant based the values of DirLat and DirLong

Algorithm 1.2 Construction of k-d tree for cloud node

An index table containing the ip addresses along with the physical location and average load degree of all registered cloud-enabled data center is maintained. Using the physical locations of the cloud-enabled data centers we have constructed a k-d tree of cloud nodes as described below.

P contains the set of geographical cloud nodes represented by their geographical coordinates.

BuildKdTree (P, depth)

1. If (P contains only one point) then
1.1.Return P
2. Else if (depth is even) then
2.1. Split P into two subsets with a vertical line *l* through the latitude closest to the mean of latitude of the coordinates in P. Let P_1 be the set of points to the left of *l* or on *l* and P_2 be the set of coordinates to the right of *l*.
3. Else
3.1. Split P into two subsets with a horizontal line *l* through the longitude closest to the mean of longitude of the coordinates in P. Let P_1 be the set of coordinates to the below of *l* or on *l* and let P_2 be the set of coordinates above *l*.
4. $\mathbf{kdTree_{left}}$ = BuildKdTree (P_1, depth + 1)
5. $\mathbf{kdtree_{right}}$ = BuildKdTree (P_2, depth + 1)
6. Create a node **kdTree** storing $\mathbf{kdTree_{left}}$ the left child of **kdTree** and $\mathbf{kdtree_{right}}$ the right child of **kdTree**.
7. Return **kdTree**.

4. Conceptual Example of Proposed Organizing Structure of Cloud-Enabled Data Center

Let us consider the practical scenario where multiple users from different geographical locations want to simultaneously access different host servers of web services (viz. Facebook, Twitter, and YouTube etc.). Based on the current load status main host server can redirect this web traffic to the different servers' host address in order to balance the total load of cloud system. In our proposed model an index table as shown in Table 1 is considered, where ip addresses along with the physical location of the different cloud-enabled data centers are enlisted. Our hierarchical k-d tree based organizing methodology of cloud server is depicted in Figure 2 and 3. In this context our main aim to establish a cloud division rule where the set of cloud nodes is partitioned using k-d tree as shown in Figure 4 and 5.

Table 1. Index Table of Registered Cloud-enabled Data Center

Considered Cloud-enabled data centers	Region of cloud-enabled data centers	IP Address of cloud-enabled data centers	Geographical coordinates of cloud-enabled data centers	
			Latitude	Longitude
A	Malaysia	202.185.107.23	2.3167 °N	111.55 °E
B	Srilanka	220.247.227.202	7 °N	81°E
C	Thailand	158.108.216.5	13.75° N	100.4833° E
D	Kolkata	203.197.118.81	22.5667 °N	88.3667 °E
E	Karachi	124.29.236.154	24.86°N	67.01°E
F	Egypt	195.246.42.198	26°N	30°E
G	Japan	133.3.250.46	35°N	136°E
H	Italy	159.149.53.216	41.9°N	12.5°E

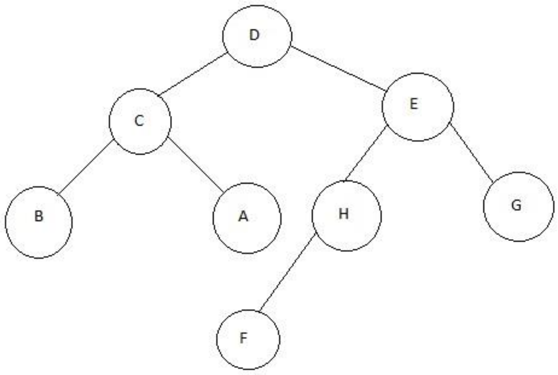

Figure 2. Constructed k-d Tree for Organizing Cloud-enabled Data Centers

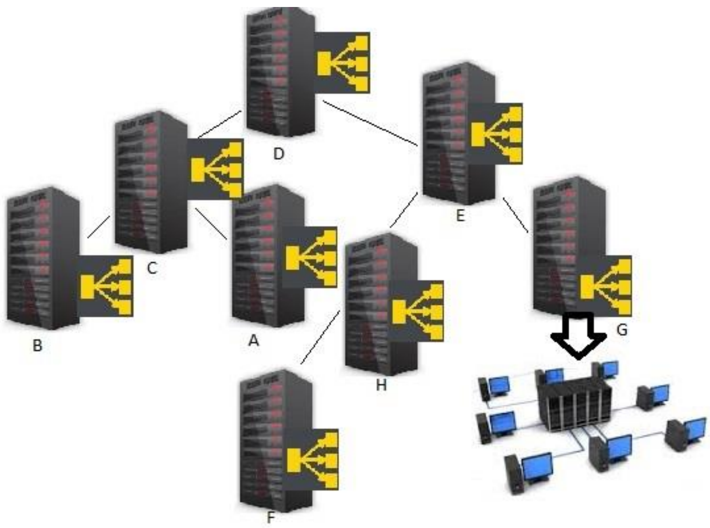

Figure 3. Detailed Structural View of Considered Cloud-enabled Data Centers

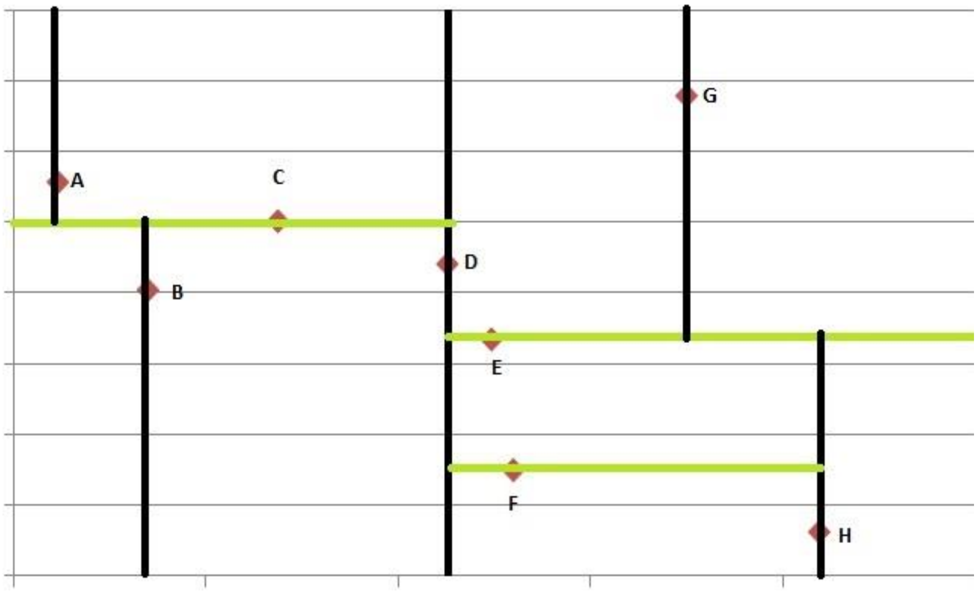

Figure 4. North-East Region Containing Considerate Cloud-enabled Data Centers is Logically Partitioned using k-d tree Technique

Figure 5. Conceptual View of our Proposed Cloud Division Methodology

5. Conclusion

Our proposed cloud-enabled data center organizing methodology is an eminent pathway of cloud division rule for various cloud-enabled data centers in the cloud system in order to achieve graceful performance. This phenomenon may leads to maximum resource utilization using various load balancing approaches. Proposed hierarchical structure can formulate an effective strategy of searching an intended data center using geographic coordinate system and redirect the traffic to the several virtual servers. In near future sketching of an efficient searching mechanism could lead this structure as a better approach for organizing the cloud- enabled data center in the cloud system.

References

[1] G. Xu, J. Pang and X. Fu, "A Load Balancing Model Based on Cloud Partitioning for the 2 Public Cloud", IEEE Xplore 2013, Vol. 18, No. 1, **(2013)**, pp. 34-39.
[2] A. D. Jadhao, M. Shetty M and D. Patil, "Comparison of Load Balancing Algorithm in Cloud Computing Environment", IJCITB, vol. 1, no. 2, **(2012)**, pp. 22-28.
[3] R. P. Padhy and G. P. Rao, "Load balancing in cloud computing system", Department of CSE, **(2011)**, NIT Rourkela, Rourkela, India.
[4] P. Jamuna and R. A. Kumar, "Optimized Cloud Partitioning Technique to Simplify Load Balancing", IJARCSSE, vol. 3, no. 11, **(2013)**, pp. 820-822.
[5] M. Katyal and A. Mishra, "A Comparative Study of Load Balancing Algorithms in Cloud Computing Environment", IJDCC, vol. 1, **(2013)**, pp. 5-14.
[6] PL. Doddini, "Load Balancing Algorithms in Cloud Computing", IJACMS, **(2013)**, vol. 4, pp. 229-233.
[7] M. Randles, D. Lamb and A. Taleb-Bendiab, "A comparative study into distributed load balancing algorithms for cloud computing", IEEE 24th International Conference on Advanced Information Networking and Applications, Perth, Australia, **(2010)**.
[8] N. Swarnkar, A. K. Singh and R. Shankar, "A Survey of Load Balancing Techniques in Cloud Computing", IJERT, vol. 2, **(2013)**, pp. 800-804.
[9] T. Desai and J. Prajapati, "A Survey Of Various Load Balancing Techniques And Challenges In Cloud Computing", IJSTR, vol. 2, **(2013)**, pp. 158-161.
[10] R. G. Rajan, V. Jeyakrishnan, "A Survey on Load Balancing in Cloud Computing Environments", IJARCCE, vol. 2, **(2013)**, pp. 2278-1021.
[11] D. Rastogi, A. Bansal, N. Hasteer, "Techniques of Load Balancing in Cloud Computing: A Survey", International Conference on Computer Science and Engineering (CSE), **(2013)**, Bangalore, India, pp. 140-143.
[12] K. S. Rashmi, V. Suma and M. Vaidehi, "Enhanced Load Balancing Approach to Avoid Deadlocks in Cloud", Special Issue of IJCA, **(2012)**, pp. 31-35.
[13] J. B. Rosenberg, "Geographical Data Structures Compared: A Study of Data Structures Supporting Region Queries", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, **(1985)**, vol. 4, pp. 53-67.
[14] R. Divya, S. P. Deepak, "A Load Balanced Greening Approach for Proficient Resource Allocation with Cloud Partitioning", IJIRCCE, **(2014)**, vol. 2, pp. 2482-2488.
[15] T. H. Cormen, C. Stein, R. L. Rivest and C. E. Leiserson, "Introduction to algorithms, 3rd ed" Cambridge, MA, USA: MIT Press, **(2009)**.

Authors

Sandip Roy, he is currently pursuing Ph.D from University of Kalyani. He is an Assistant Professor in the Department of Information Technology, Brainware Group of Institutions, Kolkata, West Bengal, India. He has completed M.Tech in Computer Science & Engineering from HIT under WBUT in 2011. He has also done his B.Tech in Information Technology from WBUT in 2008. His main areas of research interest are Cloud Computing, Data Structure and Algorithm.

Rajesh Bose, he is currently pursuing Ph.D from University of Kalyani. He is an IT professional employed as Senior Project Engineer with Simplex Infrastructures Limited, Data Center, Kolkata. He received his degree in M.Tech. in Mobile Communication and Networking from WBUT in 2007. He received his degree in B.E. in Computer Science and Engineering from BPUT in 2004. He has also several global certifications under his belt. These are CCNA, CCNP-BCRAN, and CCA (Citrix Certified Administrator for Citrix Access Gateway 9 Enterprise Edition), CCA (Citrix Certified Administrator for Citrix Xen App 5 for Windows Server 2008). His research interests include cloud computing, wireless communication and networking.

Tanaya Roy, she is currently working as an Assistant Professor in the Department of MCA, Brainware Group of Institutions. She has completed M.Tech in Computer Science & Engineering from University of Calcutta in 2011. She has honored 1st rank in her MCA from University of Calcutta in 2009. She has passed B.Sc. in Mathematics (Hons.) from St. Xavier's Kolkata under University of Calcutta in 2006. Her area of research in cloud computing and different virtualization techniques.

Debabrata Sarddar, he is an Assistant Professor in the Department of Computer Science and Engineering, University of Kalyani, Kalyani, Nadia, West Bengal, INDIA. He has done PhD at Jadavpur University. He completed his M. Tech in Computer Science & Engineering from DAVV, Indore in 2006, and his B.E in Computer Science & Engineering from NIT, Durgapur in 2001. He has published more than 75 research papers in different journals and conferences. His research interest includes wireless and mobile system and WSN, Cloud computing.

