Global Journals LaTeX JournalKaleidoscope™
Artificial Intelligence formulated this projection for compatibility purposes from the original article published at Global Journals. However, this technology is currently in beta. *Therefore, kindly ignore odd layouts, missed formulae, text, tables, or figures.*

A Tool based Edge Server Selection Technique using Spatial Data Structure

Sandip Roy[1], Debabrata Sarddar[2] and Rajesh Bose[3]

[1] W.B.U.T

Received: 14 December 2013 Accepted: 1 January 2014 Published: 15 January 2014

Abstract

Space partitioning is the process of dividing a Euclidean space into a non-overlapping regions. Kdimensional tree is such space-partitioning data structure for partitioning a Euclidean plane like the surface of earth. This paper describes a tool-based logically partitioning technique of earth surface using K-dimensional tree to segregate the edge servers over the earth surface into a nonoverlapping regions for the particular Content Delivery Network. Consequently selecting an edge server based on Least Response Time lo ad balancing algorithm is introduced to improve end-user response time and fault tolerance of the host server.

Index terms— content delivery network, K-d tree, least response time, load balancing, nearest neighbor search, spatial data structure.

1 Introduction

ontent Delivery Network (CDN) is a large distributed network of multiple data centers scattered over the earth surface [1] [3]. Today CDNs deliver a huge number of internet content including text, scripts and images and also on-demand streaming media files. Content providers pay CDN operators (e.g. Akamai, Mirror Image Internet etc.) for delivering the aforesaid contents to their customer to improve the overall network performance [2].

In this paper we have introduced a tool for partitioning earth surface using K-d Tree and also a closest edge server is selected based upon proposed least response time load balancing strategy. II.

2 Background Studies a) Content Delivery Network (CDN)

Increasing the global availability of the internet content, improving the page load time and reducing the bandwidth cost CDN edge servers are scattered over the earth surface. When users from different location are requesting for a particular web content which is algorithmically direct to the nearest edge server to achieve the goal. In this paper we have instigated a technique for partitioning earth surface using the K dimensional tree (K-d tree) and select the nearest edge server using least response time load balancing method which is discussed below.

3 b) K-dimensional Tree (K-d tree)

K-d Tree is space partitioning data structure for arranging coordinate points (latitude, longitude) over the earth surface. It can be sub-divided the earth surface into a non-overlapping regions [4] [5] [7]. In this context we have described an efficient edge server searching technique using K-d tree.

4 c) Least Response Time

The Least Response Time is a one of the most popular load balancing technique is used in this context [13] [14]. Using the aforesaid load balancing algorithm, to regulate how to dispense load among the edge servers. This paper we have used the network "ping" command to get average response time of the edge servers which are scattered over the earth surface [10].

5 III.

6 Proposed Algorithm

In our proposed algorithm we have prepared an efficient tool for CDN provider which is supervising a CDN to select low latency edge server. The set of edge servers are considered as the set of coordinate points P (e.g. latitude and longitude) scattered over geographical region and here we build a K-d tree using P which is scattered over the earth surface as shown in figure 2 and logically partition the edge servers into a nonoverlapping region as like figure 3 [6]. Using function kd_closestpointsearch, we have found nearest edge servers of the current location of end-user (e.g. Kolkata) [8] [9] [11]. Then we can calculate accurate network latency using "ping" command over the closest edge servers' IP address to find the minimum average latency time for delivering web content of a particular host server [10]. Executing "ping" command we get status(D D D D D D D D)

Year 2014 G and result information of the edge server, if the status value is 0 means server is active otherwise 1 signifies the server is dead.

Our proposed algorithm is developed using Matlab R2012b which is described below **??**11] [12]. In figure 3, the black maker is depicted that the current location of end-user and the closest edge server, among different edge servers, is waiting to send the web content to the end user that is our primary challenge.

7 Simulation analysis

Step 1 : Latitude and Longitude value of different edge servers are assigned in lat and lon array variables, which are enlisted in table 1 Step 2 : The IP Address of different location of edge servers are assigned in ipaddr variables and the IP Address of edge servers along with domain names are listed in table 3.

Step 3 : The edge servers are decomposed using K-d tree as shown in figure 3.

Step 4 : Using kd_closestpointsearch function we have search closest edge server of the current location of end-user and consequently find out the accurate network latency time using "ping" command.

Step 5 : The connection is established between least average response time active edge server located at Singapore and end-user from Kolkata for sending the web content as shown in figure 4.

8 V. conclusion

Our proposed tool and simulation results proclaim minimum network latency and minimum packet loss in selection of closest edge server over the earth surface. In this paper we have used K-d tree algorithm for decomposing the earth surface. Usage of kd_closestpointsearch method helps us to find the nearest edge server of the end-user. Our proposed tool can be used in wireless network and wired network for delivering the web content efficiently to improve the throughput of total network.

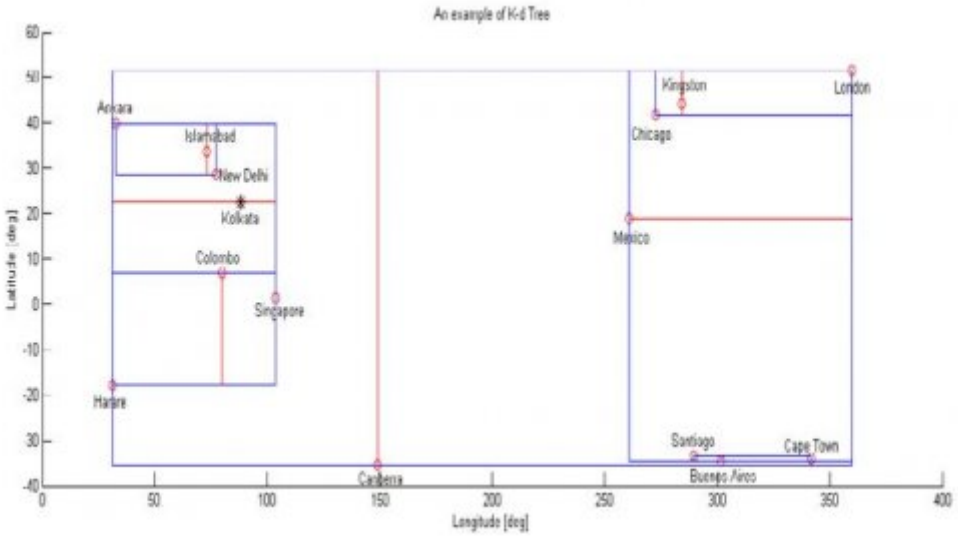
9 Global Journal of Computer Science and Technology

1

Figure 1: Figure 1 :

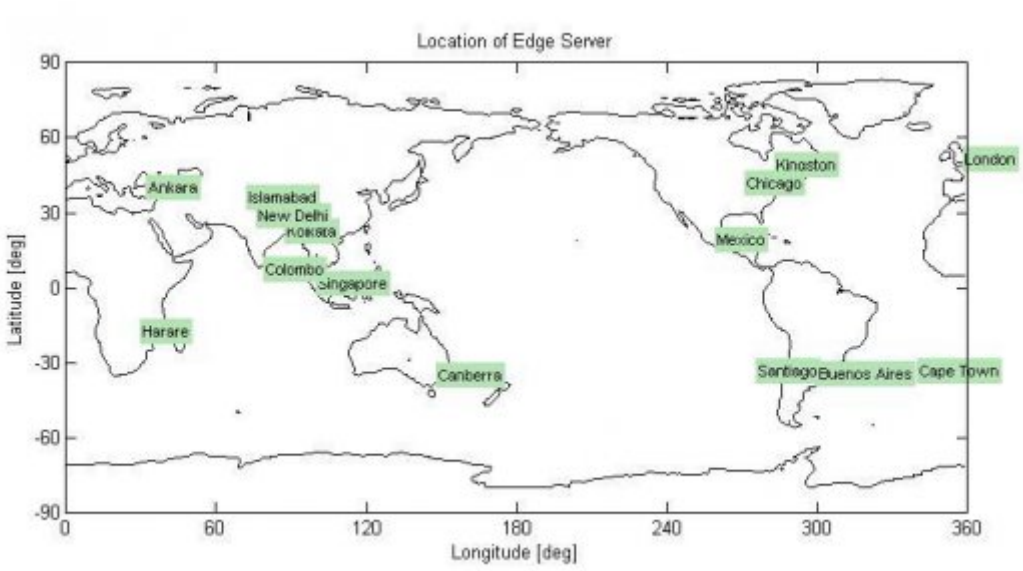
2

Figure 2: Figure 2 :



3

Figure 3: Figure 3 :



4

Figure 4: Figure 4 :

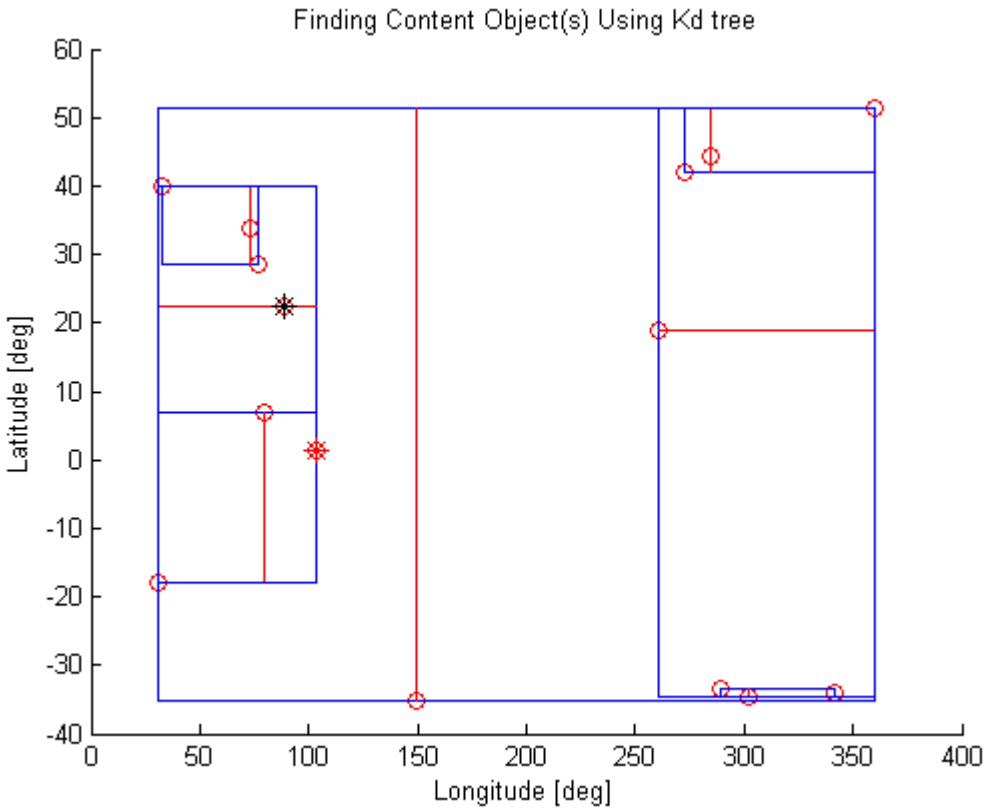


Figure 5: Volume

1

Year 2014
15
Volume XIV Issue III Version I
D D D D D D D D) G
(
Global Journal of Computer Science and Technology

Location of edge server	Latitude	22.5667°N	Longitude	88.3667°E
Kolkata	1.3000°N	6.9344°N	103.8000°E	79.8428°E
Singapore Colombo	51.5072°N	41.8819°N	0.1275°W	87.6278°W
London Chicago	28.6139°N	39.9300°N	77.2089°E	32.8600°E
New Delhi Ankara	33.7167°N	33.4500°S	73.0667°E	70.6667°W
Islamabad Santiago	19.000°N	44.2333°N	99.1333°W	75.6919°W
Mexico Kingston	34.6033°S		58.3817°W	
Buenos Aires				
Harare	17.8639°S		31.0297°E	
Cape Town	33.9253°S		18.4239°E	
Canberra	35.3075°S		149.1244°E	

Figure 6: Table 1 :

2

Location of Edge Servers	Latitude	Longitude	Modified Longitude
Kolkata	22.5667	88.3667	88.3667
Singapore	1.3000	103.8000	103.8000
Colombo	6.9344	79.8428	79.8428
London	51.5072	-0.1275	359.8725
Chicago	41.8819	-87.6278	272.3722
New Delhi	28.6139	77.2089	77.2089
Ankara	39.9300	32.8600	32.8600
Islamabad	33.7167	73.0667	73.0667
Santiago	-33.4500	-70.6667	289.3333
Mexico	19.0000	-99.1333	260.8667
Kingston	44.2333	-75.6919	284.3081
Buenos Aires	-34.6033	-58.3817	301.6183
Harare	-17.8639	31.0297	31.0297
Cape Town	-33.9253	-18.4239	341.5761
Canberra	-35.3075	149.1244	149.1244

Figure 7: Table 2 :

3

Location of Edge Servers	IP Address	Domain Name
Kolkata	203.197.118.81	www.jaduniv.edu.in
Singapore	137.132.21.27	www.nus.edu.sg
Colombo	192.248.17.88	www.cmb.ac.lk
London	212.113.11.22	www.lse.ac.uk
Chicago	198.101.129.15	www.uchicago.edu
New Delhi	103.27.9.20	www.du.ac.in
Ankara	80.251.40.153	www.ankara.edu.tr
Islamabad	61.5.158.124	www.islamabadairport.com.pk
Santiago	158.170.64.116	www.udesantiago.cl
Mexico	128.123.3.2	www.nmsu.edu
Kingston	130.15.126.136	www.queensu.ca
Buenos Aires	190.224.163.234	www.buenosairesherald.com
Harare	196.201.17.237	www.caaz.co.zw
Cape Town	41.72.141.237	www.capetown.travel
Canberra	137.92.97.88	www.canberra.edu.au

Figure 8: Table 3 :

4

Location of Edge Servers	IP Address	Status	Average time(ms)
Canberra	137.92.97.88	Dead	Request timed out
Kolkata	203.197.118.81	Active	260
Colombo	192.248.17.88	Active	Destinationhost unreachable
Singapore	137.132.21.27	Active	238

Figure 9: Table 4 :

71 [1] [2]

[1] © 2014 Global Journals Inc. (US)Response Time load balancing algorithm is introduced to
[2] © 2014 Global Journals Inc. (US)

[Patel et al. ()] , V P Patel , H D Patel , J P Patel . *A Survey on Load Balancing in Cloud Computing. IJERT* 2012. (9) p. 1.

[Friedman et al. ()] 'An Algorithm for Finding Best Matches in Logarithmic Expected Time'. J H Friedman , J Bentely , R A Finkel . *ACM Transactions on Mathematical Software* 1977. 3 p. .

[Sarddar et al. ()] *An Efficient Edge Servers Selection in Content Delivery Network Using Voronoi Diagram*, D Sarddar , S Roy , R Bose . 2014. IJRITCC p. .

[Moore] *An introductory tutorial on KD trees*, A Moore .

[Vaidya ()] 'An O(n log n) Algorithm for the All-Nearest-Neighbors Problem'. P M Vaidya . doi:10.1007/BF02187718. *Discrete and Computational Geometry* 1989. 4 (1) p. .

[Clarkson ()] 'Fast algorithms for the all nearest neighbors problem'. K L Clarkson . 10.1109/SFCS.1983.16. *24th IEEE Symp. Foundations of Computer Science, (FOCS '83)*, 1983. p. .

[Rosenberg ()] 'Geographical Data Structures Compared: A Study of Data Structures Supporting Region Queries'. J B Rosenberg . 10.1109/TCAD.1985.1270098. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 1985. 4 (1) p. .

[Parikh et al. ()] 'Globally Distributed Content Delivery'. J Parikh , H Prokop , R Sitaraman , J Dilley , B Maggs , B Weihl . *IEEE INTERNET COMPUTING* 2002. p. .

[Mata-Toledo and Gupta ()] 'Green data center: how green can we perform'. R Mata-Toledo , P Gupta . *Journal of Technology Research* 2010. Academic and Business Research Institute. 2 (1) p. .

[Chandran] *Introduction to kd-trees*, S Chandran . University of Maryland Department of Computer Science

[Vemulapalli ()] *Kd tree implementation in Matlab*, P Vemulapalli . `http://www.mathworks.in/matlabcentral/fileexchange/26649-kdtree-implementation-in-matlab` 2010. (Retrieved from MATLAB CENTRAL website)

[Kherani and Vania ()] *Load Balancing in cloud computing*, F F Kherani , J Vania . 2014. IJEDR p. .

[Bentley ()] 'Multidimensional binary search trees used for associative searching'. J L Bentley . 10.1145/361002.361007. *Communications of the ACM* 1975. 18 (9) p. 509.

[Repantis et al. ()] 'Scaling a Monitoring Infrastructure for the Akamai Network'. T Repantis , J Cohen , S Smith , J Wein . *ACM SIGOPS Operating Systems Review* 2010. 44 (3) p. .

[Nygren et al. ()] 'The Akamai Network: A Platform for High Performance Internet Applications'. E Nygren , R K Sitaraman , J Sun . *ACM SIGOPS OSR* 2010. 44 (3) p. .