www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 6 Issue 5 May 2017, Page No. 21196-21200
Index Copernicus value (2015): 58.10 DOI: 10.18535/ijecs/v6i5.01

Obtaining Good Heuristics for Big Data Broadcasting Problem Using Novel Pipeline Approach.

Prof. Priyadarshani Kalokhe, Abhishek Kumbhar, Jyoti Shelke, Rinkal Bari, Snehal Shewale
hodcomp.acem@gmail.com
abhi6034@gmail.com
jyotishelke107@gmail.com
rinkalbari19@gmail.com
snehals2424@gmail.com

Abstract – *Today big data computing is a critical challenge faced by many industries .Everyday up to terabytes to petabytes of the data is generated in the cloud computing paradigm. Thus there is a drastic demand for building a service to distribute, manage and process massive data sets that has risen tremendously. Engineers and researchers are dealing with these large dataset. In this paper, we investigate the big data broadcasting problem for single source node to broadcast a big chunk of data to a set of nodes with the objective of minimizing the maximum completion time. These nodes may be present in the same data center or across geo distributed data centers. We model the big data broadcasting problem into a* ***LOCKSTEP BROADCAST TREE (LSBT).****The main idea of this model is to define a basic unit of upload bandwidth r, such that a node with capacity c broadcast a data to a set of nodes at the rate r. We divide the broadcast data into m chunks. These data chunks can then be broadcasted using a pipeline approach.*

KEYWORDS: *Big data computing, delivery algorithm, cloud computing.* -

1. INTRODUCTION

Today the major efforts are being taken on big data computing. The advances in the scommunication and storage have resulted in enormous data sets in business, science and the society being generated and analyzed to explore the value of those data. Today engineers and researchers are dealing with terabytes and petabytes of datasets in cloud computing paradigm. For example, the industries such as Google, Yahoo, and Amazon collect huge amount of data every data for proving information services freely to people in useful ways. Thus the demand for building a distributed service stack to efficiently distribute, manage and process massive datasets has risen drastically. Earlier there were several efficient techniques proposed to manipulate huge amount of data ranging from terabytes to petabytes of data on as many as thousands of machine.

There are lot of application domains which widely apply broadcasting operations such as scientific data distribution, database transaction logs backup, latest security patches, multimedia streaming applications among distributed data centre. Therefore we focus on big data broadcasting operation which is one of the most essential communication mechanism distributed system. For big data computing data transmission overhead is a significant factor of a job completion time. It is shown that the total amount of data transmission time occupies approximately one third of the jobs running time in the hadoop tracing logs of Facebook.

1.1 Potential application of LSBT

There are at least three broad applications where LSBT can be applied:1)topology control in BitTorrent like systems; 2) data broadcasting in cloud computing software stack; 3) energy

conservation in peer-assisted content delivery services. We consider these in the context of network systems that are heterogeneous network environments. LSBT could be useful in topology control in BitTorrent-like systems. BitTorrent is a peer-to-peer application which aims to enable the fast and efficient distribution of large files among a large group of nodes. In BitTorrent, each peer maintains a constant number of concurrent upload connections (usually five).

LSBT can be integrated into the cloud computing software stack. For example, Apache Hadoop1 is a software framework that allows for the distributed processing of large data sets across clusters with thousands machines. Thus an efficient and scalable way to disseminate a large volume of data among machines is a significant challenge in Hadoop [8]. Our LSBT can be integrated into the delivery services of Open Stack software stack.

2. Related work

It focus on the big data broadcasting operation that is one of the most essential communication mechanisms in distributed systems. There are a lot of application domains that widely apply broadcasting operations, such as scientific data distributions [9], database transaction logs backups, the latest security patches, multimedia streaming applications, and data replica or virtual appliance deployment [3] among distributed data centers. Since the size of data becomes so enormous, the impact of broadcasting operation also becomes increasingly significant. We introduce the novel LockStep Broadcast Tree (LSBT) to model the Big Data .

2.1 MapReduce: A Simplified Data Processing on Large Clusters

MapReduce is a programming model and an associated implementation for processing and generating large data sets. Users specify a map function that processes a key/value pair to generate as set of intermediate key/value pairs, and a reduce function that merges all intermediate values associated with the same intermediate key. Many real world tasks are expressible in this model. Programs written in this function a style are automatically parallelized and executed on a large cluster of commodity machines. The run-time system handles the details of partitioning the input data, scheduling the program's execution across a set of machines, handling machine failures, and managing the required inter-machine communication. This allows programmers without any experience with parallel and distributed systems to easily utilize the resources of a large distributed system.

Advantage:

This technique is simple and powerful. Interface that enables automatic parallelization and distribution of large-scale computations, combined with an implementation of this interface that achieves high performance on large clusters of commodity PCs.

2.2 Data Replication in Data Intensive Scientific Applications with Performance Guarantee

We demonstrate that the distributed caching technique significantly outperforms an existing popular file caching technique in Data Grids, and it is more scalable and adaptive to the dynamic change of file access patterns in Data Grids. Replication is an effective mechanism to reduce file transfer time and bandwidth consumption in Data Grids placing most accessed data at the right locations can greatly improve the performance of data access from a user's perspective

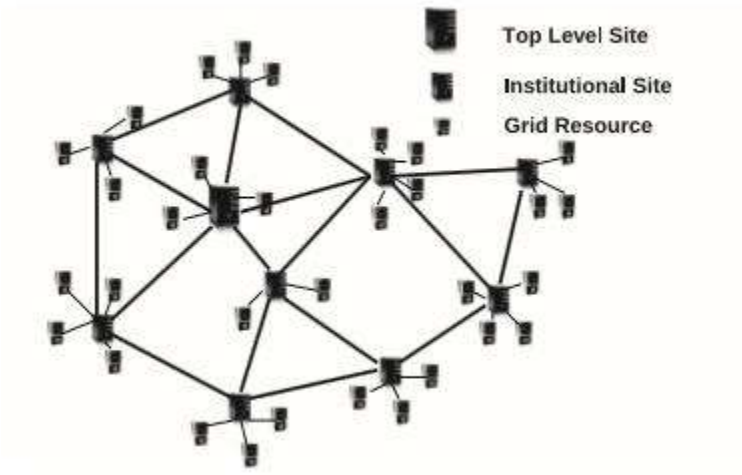


Fig 1 – Peer to peer connectivity.

2.3 Analyzing and Improving a Bit Torrent Network's Performance Mechanisms

We find Bit Torrent to be remarkably robust and scalable at ensuring high uplink bandwidth utilization. It scales well as the number of nodes increases, keeping the load on the origin server bounded, even when nodes depart immediately after completing their downloads. The LRF policy performs better than alternative block-choosing policies in a wide range of environments (e.g., flash crowd, post-flash crowd situations, small network sizes, etc.) By successfully getting rid of the last block problem, it provides a simpler alternative to previously proposed source coding strategies, e.g. Second, we find that Bit Torrent shows sub-optimal behavior.

3.Equations

PHASE 1: NW CONFIG.

- $ki = |ci / r|$

Where k is number of edges.

- $r = arg\ min\ D(c, r) = arg\ min \sum B/ r$

Where r is uploading rate and D is maximum completion time.

- $c = r * k$

Where c is uploading capacity

PHASE 2: Centralized Heuristic Solution

- $D = (B/r)log\ k\ n = (kB/c)(ln\ n / ln\ k)$

Where B is a size of data chunks.

4. Algorithm

Here we model a Lock Step Broadcast Tree (LSBT) which will define an upload bandwidth(r) and each node capacity(C) at c/r per children. After these data sets divided into chunks, and are broadcasted in a pipeline manner. In homogeneous network environment the capabilities of each node is C, and uplink rate r which gives less maximum completion time. In heterogeneous networks capabilities C1, C2...Cn, so optimal uplink rate r will be calculated by O (nlog2n) algorithm there after we construct a LSBT. Finally our results show less computational complexity and less maximum completion time over other broadcasting techniques.

First we use algorithm 1 to find out the candidate set based on capabilities C, and upper bound and lower bound values of r.

There after we implement algorithm 2 for optimal LSBT based on r values.

Algorithm 1 Discretization algorithm for the candidateset

Input: a set of upload capacities c and the upper and lower bounds of r

BEGIN UnionSet ← empty

CandidateSet ← empty

for i ← 1 to n do

UnionSet ← UnionSet

∪ ci

end

for UnionSet ← Sort(UnionSet)

for j ← 1 to n − 1 do

for all u in UnionSet

do r ← u=k if r ≥ upper then

Continue

Else if r < lower then

Break

End if

CandidateSet ← CandidateSet

∪ r End for

End for

Return

CandidateSet END

Output: CandidateSet After implementing candidate set we find out bandwidth r n maximum completion time D by using below

DOI: 10.18535/ijecs/v6i5.01

Algorithm 2 The r -search algorithm for the optimal LSBT

Input: a set of upload capacities c and CandidateSet

BEGIN CandidateSet ← Sort(CandidateSet)

D← ∞ r← empty

for h ← 1 to 2(⌊log2 n⌋ + 1)

do rt ← 1 lt ← Sizeof(CandidateSet)

while rt ≤ lt do mid ← ⌊(rt + lt)=2⌋

r ← CandidateSet[mid] t ← GLSBT(c; r)

if lt − rt = 1 and t:Height = h then

d ← t:BroadcastingTime

if d < D then r← r D← d

end if

else

break

end if

if h ≤ t:Height then

rt ← mid

else lt ← mid − 1

end if

end while

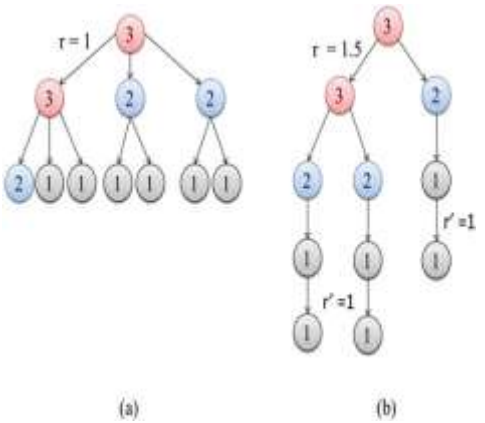
end for

return r and D

END

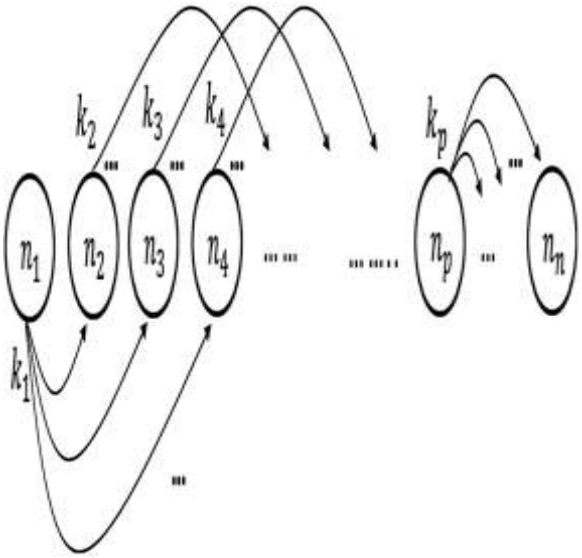
Output: r and D

5. Figures and Tables



The two examples of LSBT. The tree (a) presents the optimal LSBT with r_ ¼ 1, the maximum completion time D is two units of time and tree (b) requires three units of time. Assume that the size of data chunk B ¼ 1 and the digits specify these node's upload capacity.

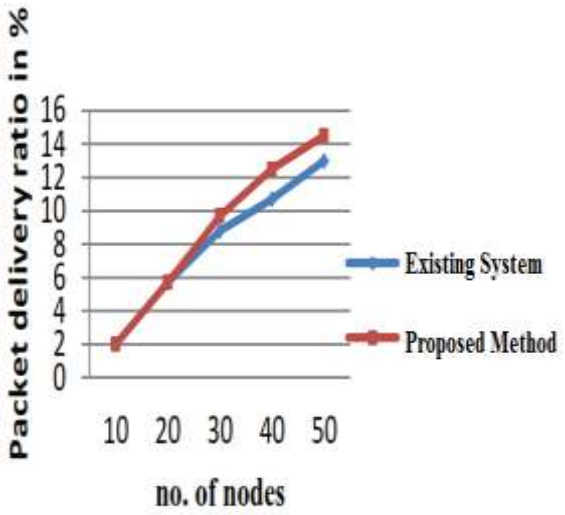
Suppose that we have a set of n nodes, and k1; k2; k3; . . . ; kn are the number of edges of each node.



6. Result & Analysis

The performance of our LSBT by using numerical analysis. Here we implemented two approaches .one is FNF approach and another is LSBT approach. This analyse algorithm as the number of nodes in network increases the size of candidate set slowly vary. Compared with naive approach our approach gives good performance and this gives good heuristic for

reducing candidate set. We now show the maximum completion time for two algorithms. We select the networks with the number of nodes (n) 100,1000,10000 nodes. Size of file is 50 mb and chunks are 100. Number of nodes versus maximum completion time is proposed. As going on increasing no of nodes the maximum completion will reduce. Proposed Method the computation time as number of nodes increases.



7. Conclusion & Future Work

We conclude that from the different algorithms and systems we learnt how efficiently we can broadcast the data ,share the data in centralized system, distributed system, peer 2 peer networks. We have introduced a framework that is prepared to do naturally extricating information parallelism from synchronising applications. Our work contrasts from former work by having the capacity to concentrate such parallelism with wellbeing sureties in the vicinity of administrators that can be stateful , particular, and client denied. We have exhibited that these procedures can scale with accessible assets and exploitable parallelism.

8. Referances

[1] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," Proc. Oper. Syst. Design Implementation, 2004, pp. 137–150.

[2] D. Nukarapu, B. Tang, L. Wang, and S. Lu, "Data replication in data intensive scientific applications with performance guarantee," IEEE Trans. Parallel Distrib. Syst., vol. 22, no. 8, pp. 1299– 1306, Aug. 2011.

[3] M. Banikazemi, V. Moorthy, and D. Panda, "Efficient multimedia broadcast for heterogeneous susers in cellular networks " in Combinatorial Algorithms. New York, NY, USA: Algorithmics Press, 1972

[4] M. Chowdhury, M. Zaharia, J. Ma, M. I. Jordan, and I. Stoica, "Managing data transfers in computer clusters with orchestra," Proc. ACM Special Interest Group Data Commun., 2011, pp. 98–109.

[5] A. R. Bharambe, C. Herley, and V. N. Padmanabhan, "Analyzing and improving a bittorrent networks performance mechanisms," in Proc. IEEE Int. Conf. Comput. Commun., 2006, pp.

[6] O. Beaumont, L. Eyraud-Dubois, and S. K. Agrawal, "Broadcasting on large scale heterogeneous platforms under the bounded multi-port model," in Proc. IEEE Int. Symp. Parallel Distrib. Process. Symp., 2010, pp. 1–11.

[7] K.-S. Goetzmann, T. Harks, M. Klimm, and K. Miller, "Agent based approach for modeling peer-to-peer networks," Proc.

[8] L. Massoulie, A. Twigg, C. Gkantsidis, and P. Rodriguez, "P2P streaming capacity under node degree bound," IEEE 30th Int. Conf. Dist. Comput. Syst., pp. 587–598, 2010.

[9] R. Thommes and M. Coates, "Bittorrent fairness: Analysis and improvements," in Proc. Workshop Internet, Telecommun. Signal Process., Dec. 2005, pp. 9–14.

[10] O. Beaumont, L. Eyraud-Dubois, and S. K. Agrawal, "Broadcasting on large scale heterogeneous platforms under the bounded multi-port model," in Proc. IEEE Int. Symp. Parallel Distrib. Process. Symp., 2010, pp. 1–11.